

QL Today

Volume 9
Issue 6
March/April
2005

ISSN 1432-5454

The Magazine about QL, QDOS,
Sinclair Computers, SMSQ...

EASY POINTER Vers. 4 - now supporting scaleable windows

The screenshot displays the EASY POINTER software interface. At the top left, there are keyboard shortcuts: ESC Quit, Erase, Zz, and Mode. A logo for 'EASY POINTER' is visible. A menu is open with options: Files, Outline, Info window, Info object, Loose item (highlighted), Appl. window, Set up, and Remove. To the right, a window titled 'LI1 LI2 INFO1 LI3 LI4' is shown with 'OK' and 'Cancel' buttons. At the bottom, a table lists menu items with their coordinates and objects.

No.	X	Y	X0	Y0	Object				
1	0	18	0	10	0	2	0	1	LI1
2	0	18	0	10	1	42	0	1	LI2
3	0	18	0	10	3	140	0	1	LI3
4	0	18	0	10	4	180	0	1	LI4

WWW@

Internet on the QL

Phoebus R. Dokos starts a new series

Contents

- 3 Editorial
4 News
8 QL Software on the Web - Update
9 Helpline
10 Big Bangs and After-Glows
Geoff Wicks
11 Gee Graphics! (on the QL?) - Part 42
H. L. Schaaf
13 Letter Box
15 Scaling in EasyPTR 4 Dilwyn Jones
22 Keyboards - the missing bit
George Gwilt
23 Internet on the QL - Part I (Emulators)
Phoebus R. Dokos
32 Start Here
Essential Information For 21st Century
QLers - Part II: Hard Drives Roy Wood
38 Program Development in Assembly
Language - Part 14 Norman Dunbar
39 EASYPTR and the New Window Manager
Dilwyn Jones
49 Updates at/from J-M-S Jochen Merz
49 Interpolating Curves - Vector Graphics
Stephen Poole
54 Al Boehm
55 Bytes of Wood Roy Wood

Advertisers

in alphabetical order

Jochen Merz Software	51
QBranch	34, 35
Quanta	43
RWAP	25
TF Services	17
Geoff Wicks	9

QL Today

ISSN 1432-5454

Please visit www.QLToday.com

German office & Publisher:

Jochen Merz Software Tel. +49 203 502011
Kaiser-Wilhelm-Str. 302 Fax +49 203 502012
47169 Duisburg email: JMerz@j-m-s.com
Germany email: QLToday@j-m-s.com

English office:

Q Branch Tel. +44 1273 386030
20 Locks Hill Mobile +44 7836 745501
Portslade Fax +44 1273 381577
BN41 2LB email: qbranch@qbranch.demon.co.uk
United Kingdom email: QLToday@j-m-s.com

Editor:

Dilwyn Jones email: dilwyn.jones@tesco.net
41 Bro Emrys email: QLToday@J-M-S.com
Tal-Y-Bont, Bangor
Gwynedd
United Kingdom LL57 3YT

Co-Editor:

Bruce Nicholls Tel +44 1708 510764
38 Derham Gardens Fax +44 870 0568755
Upminster email: qltoday@q-v-d.demon.co.uk
Essex RM14 3HA email: QLToday@j-m-s.com
United Kingdom

QL Today is published bi-monthly, our volume begins on beginning of June. Subscriptions begin with the current issue at the time of sign up. Please contact the German or English office for current subscription rates.

We welcome your comments, suggestions and articles. YOU make **QL Today** possible. We are constantly changing and adjusting to meet your needs and requirements. Articles for publication should be on a 3.5" disk (DD or HD) or sent via Email. We prefer ASCII, Quill or text87 format. Pictures may be in _SCR format, we can also handle GIF or TIF or JPG. To enhance your article you may wish to include Saved Screen dumps. PLEASE send a hardcopy of all screens to be included. Don't forget to specify where in the text you would like the screen placed.

**The deadline for Issue 1 of Vol. 10
is the 15th of May!**

QL Today reserves the right to publish or not publish any material submitted. Under no circumstances will **QL Today** be held liable for any direct, indirect or consequential damage or loss arising out of the use and/or inability to use any of the material published in **QL Today**. The opinions expressed herein are those of the authors and are not necessarily those of the publisher.

This magazine and all material within is © copyright 2005 Jochen Merz Software unless otherwise stated. Written permission is required from the publisher before the reproduction and distribution of any/all material published herein. All copyrights and trademarks are hereby acknowledged.

The focus of my QL activities this month has been the major update to that widely used QL programming package, Easyptr. After much pressure was put on him to undertake the task, Marcel Kilgus finally agreed to take on the task of updating what is probably the most widely used of software writing packages. He has added new facilities like access to the System Palette colours of the new Window Manager 2 and scalable menus. Whether you use Easyptr or not, I hope that this tool can now be used to write modern QL software able to take advantage of the new colours and other features of the modern SMSQ/E.

Alongside the Easyptr update, I found a program called QCoCo (QL Colour Configurator) very useful. Wolfgang Uhlig wrote this program to help with creating System Palette colour themes. It is not until you start to use this program that you realise how useful and important that it is – reading the Window Manager 2 documentation and all the articles in QL Today might have helped, but nothing beats getting your hands dirty with subjects like this. Apart from being a useful tool in its own right, QCoCo helped make sure I fully understood the subject more quickly than I would have done otherwise. So I decided to write articles about Easyptr and QCoCo. If Jochen manages to fit them all in this issue, you'll get to read them all at once, otherwise one of them will, I'm afraid, have to wait for the next issue.

Amid all this, Per Witte released one of the most advanced games yet for the new systems. D-Miner is a Minesweeper clone for high-end SMSQ/E systems. Even Marcel Kilgus was so impressed with his that he was driven to write to describe what he saw: "I thought this thing is freakishly impressive and is really pushing the boundaries on what can be done with the PE." Compliment indeed from a person so closely involved with the development of the operating system itself.

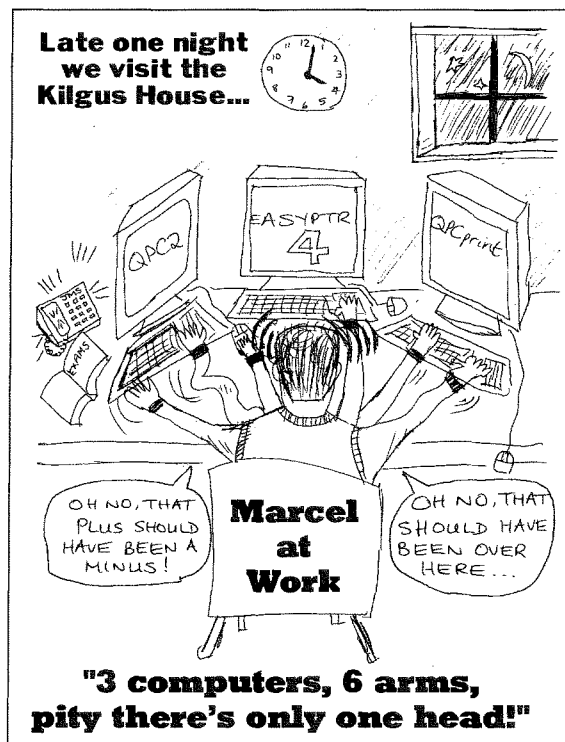
Less trumpeted was the news from Daniele Terdina of the release of another QL emulator. QemuFast is a derivative of the well known QemuLator. It is probably the fastest QL emulator yet, but the speed comes at the cost of compatibility.

QemuLator is well known for its software compatibility, but QemuFast focuses on speed. If a particular program won't run, that's that. At the moment, it's an early release but already showing great promise.

News also came (rather quietly) of a major update to Turbo. George Gwilt has been steadily updating the compiler and now he is releasing test versions able to use reference parameters, which has always been a bit of a sticking point preventing more widespread use of Turbo as it caused difficulty with compiling pointer driven applications – these problems may now disappear and we may see Turbo as becoming a real alternative to QLiberator for producing modern new QL software. Time will tell.

2005 has a busy events calendar, what with the special events, "QL 2004 After-Glow" show in The Netherlands, "QL is 21" in Britain and the other Quanta Workshops, USA show and so on. Make an effort to visit the Quanta AGM if you can, there are some possibly controversial changes to the constitution on the cards. Make sure you vote on the changes, whether you are for or against them.

Don't forget that there will be a longer than usual wait for your next QL Today with the next issue as indicated in the last issue.



Cartoon

NEWS

EASYPTR 4

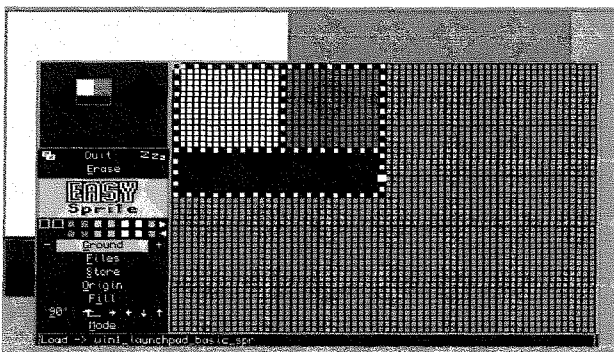
With the ever expanding scope of SMSQ/E it became clear that an update to one of the most widely used QL programming packages was sorely needed.

Marcel Kilgus was given permission by the original author, Albin Hessler, to update the package and has put quite a lot of work into producing Easyprtr 4, due to be released at about the time you read this.

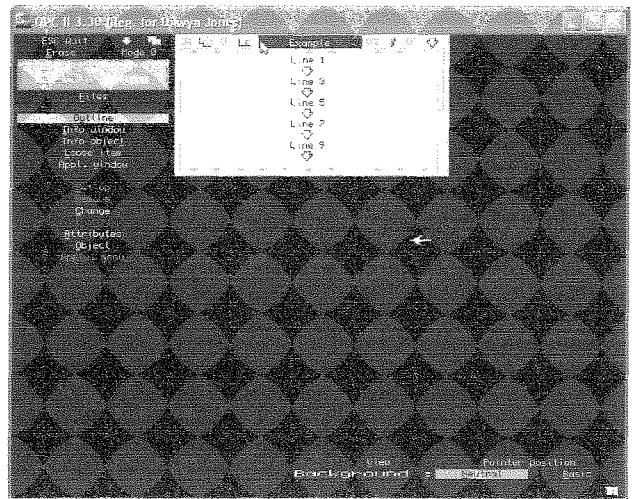
To help with development, Marcel set up a 'testers group' who got several updated versions in quick succession and feedback from these early users was used to fix bugs and add some new features. The actual updates list is quite extensive and even Marcel is quick to point out that even he cannot remember everything that's been changed!

The main changes are support in Easymenu for the System Palette colour schemes of Window Manager 2 and scalable menus. There are also changes such as the startup size of the Easymenu display, the MODE button in high colour cycles through the four possible system palettes and an inbuilt palette (so you can get a preview of how your program menus will look) plus better support for high colour sprites (such as those produced by Jerome Grimbert's Sprite Editor) in menus created with Easymenu. Marcel has done an update to Easysprite as well, to make it work more reliably in a GD2 environment, although it can still only produce mode 4 and mode 8 sprites - these will still display correctly in high colour modes of course.

The hope is of course that QL programmers will use this new version (which will be available from Marcel's usual software suppliers such as J-M-S.) to produce software which makes best use of the facilities available to us under SMSQ/E.



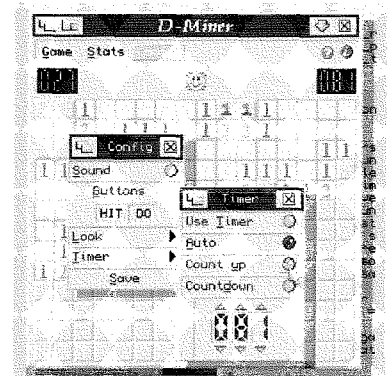
The new Easysprite



The new Easymenu

D-MINER

Per Witte was quickly out of the starting block with a Minesweeper style game for modern QL systems. Not only is it graphically modern (pushes the boundaries of QL software a bit) and the sound is brilliant (on systems supporting QLSSS - it uses .ub files played with Wolfgang Lerner's Sound Extensions) it actually plays a good game too!



I was fortunate enough to be sent an early trial version of this game and pretty good it was too. My only problem was lack of enough time to play enough to get addicted to it!

More information from Per's website at:
<http://knoware.mysite.freemove.com>

TURBO 4c21

A beta test version of Turbo compiler v4c21 is just available from John Sadler's SQLUG website at the time of writing. George Gwilt has been working on what many saw as perhaps Turbo's one major restriction, namely that parameters to procedures were passed by value, not by reference, unless you explicitly used the REFERENCE directive to bypass this, which of course needed specific changes to programs in order to be compiled correctly. Turbo 4c21 now allows you to pass parameters by reference in the normal way. The second testing version (v4b21) of Turbo corrected a fault in the earlier version which occurred with a program using MANIFEST. The ver-

sion 4b21, as did the previous one, allows parameters to be passed by "reference" so that new values can be set by the machine code routine. It also allows the passing of arrays to machine code procedures and functions.

George Gwilt would be glad of any feedback so that a final version can be produced.

ELECTION

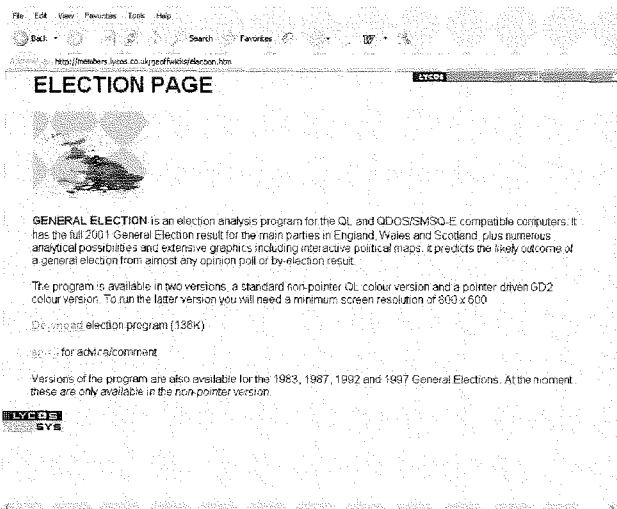
Oh no, politics even finds its way to the QL scene! Geoff Wicks writes:

"As the UK election campaign now seems to have started (election date set at 05/05/05, now there's a magic number!), I have reinstalled my election analysis program on my web site.

New is a pointer driven GD2 colour version of the program. To run this you will need a minimum screen resolution of 800 x 600.

<http://members.lycos.co.uk/geoffwicks/election.htm>

The haters of politics will be pleased to know that as soon as I had uploaded the page, Lycos attached a paint-balling advert to it. Who will take the challenge on? It would beat throwing eggs at John Prescott any day!"



Election program web page

(For anyone not aware of what Geoff is referring to, the British deputy prime minister John Prescott has been the victim of foods thrown at him and got into fisticuffs with protesters over the years!)

QL TODAY INDEX

Thanks to Brian Kemmett, I have added a set of index files for QL Today volumes 5-8 onto my website. This adds to the existing indexes for volumes 1-4. Brian has kindly offered to try to add an index to volume 9 when complete.

These are available in both plain text and Word .DOC file versions for you to print off.

The web page address is:

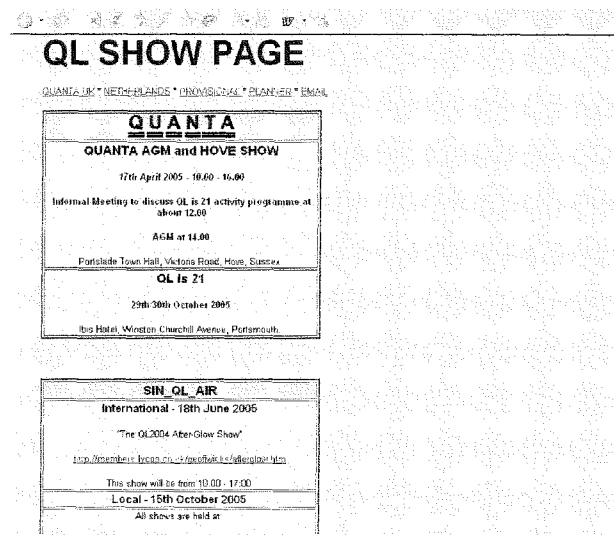
<http://homepages.tesco.net/dilwyn.jones/qltoday/qltoday.html>

QL SHOWS WEBSITE

Geoff Wicks writes:

I have now added a QL shows page to my website:

<http://members.lycos.co.uk/geoffwicks/qlshow.htm>



This gives a list of already planned shows, plus the most likely dates of other shows. You can also download the QL shows planner as a pdf document. The planner will be updated bimonthly to co-incide with the copy date of the Quanta Magazine and will probably also go onto the Quanta website.

Please keep in touch if you are planning a show, and please send me details of any website about your show as I can then add a link on my shows page.

There are two shows in which I have a special interest:

EINDHOVEN 18th JUNE

Sjef vd. Molegraaf has given this the title of the "QL2004 After-Glow Show". Our idea is that one of the great successes of QL2004 was to get most of the QL Developers together. This seems to have stimulated a lot of further development activity. Eindhoven is an easy centre for most developers to come together and we would like to repeat the experience although on a smaller scale this year. It would be a good chance to come together to brainstorm and meet socially. How many people would be interested? What would you like as activities?

PORTSMOUTH 29th/30th OCTOBER

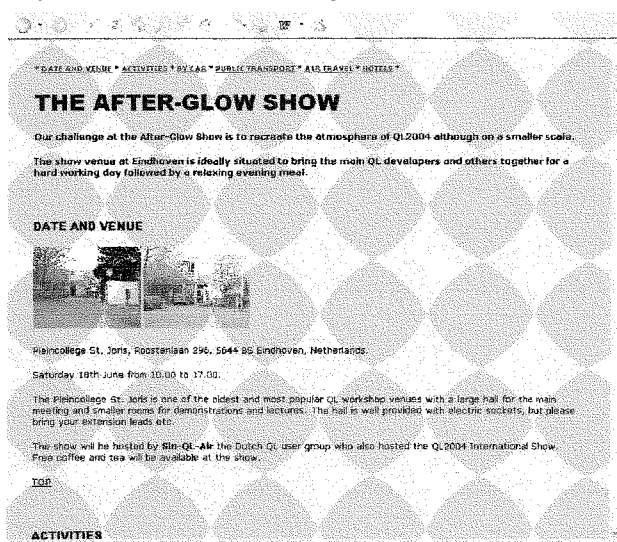
This is this year's big one and Quanta has a top notch team to run it. Roy Brereton of QL2000 fame is the General Organiser, Geoff Wicks of QL2004 fame is the Programme Organiser and Sarah Gilpin of Manchester show fame will organise the "Ladies" programme.

I have several ideas for the activity programme that I am keeping under wraps until the Hove Show. At Hove we intend to have an informal meeting (i.e. not part of the formal AGM) to discuss what members and non-members would like to see at Portsmouth. We are also happy to accept ideas via email, snailmail or telephone from members and non-members of Quanta who cannot make it to Portsmouth.

I hope to describe ideas for both shows in more detail in the next QL Today and also to set up a Portsmouth webpage shortly after the Hove AGM.

We have set up a webpage for the Eindhoven show in June. The page is already online.

<http://members.lycos.co.uk/geoffwicks/afterglow.htm>



DATE AND VENUE

THE AFTER-GLOW SHOW

Our challenge at the After-Glow Show is to recreate the atmosphere of QL2004 although on a smaller scale.

The show venue at Eindhoven is ideally situated to bring the main QL developers and others together for a hard working day followed by a relaxing evening meal.

DATE AND VENUE

Pleincollege St, Joris, Roostenlaan 296, 5644 BS Eindhoven, Netherlands.

Saturday 10th June from 10.00 to 17.00.

The Pleincollege St. Joris is one of the oldest and most popular QL workshop venues with a large hall for the main meeting and smaller rooms for demonstrations and lectures. The hall is well provided with electric sockets, but please bring your extension leads etc.

The show will be hosted by 6in-QL-Air the Dutch QL user group who also hosted the QL2004 International Show. Free coffee and tea will be available at the show.

TOP

ACTIVITIES

AfterGlow web page

We also hope to have a webpage for QL is 21. I hope to be able to have this online shortly after the Quanta AGM in April

SBASIC/SUPERBASIC REFERENCE MANUAL

Rich Mellor writes:

At long last and thanks to a special filter developed by Marcel Kilgus, I have been able to convert the whole of the SBASIC/SuperBASIC Reference Manual to Adobe Acrobat format.

This means that it is now available to buy on CD at a cost of £20. The support programs and files will still be provided on floppy disk together with the CD.

If anyone has a paper copy and would like to get the CD copy instead, then there could be a market for second hand copies of the original printed version, amongst those who do not have a PC to read the CD-ROM.

EPSON PRINTERS

Rich Mellor and Derek Stewart now have the following second hand printers for sale, all QL compatible unless otherwise stated:

26 x Epson 850 Colour Inkjet printers (this is what Rich Mellor uses)

1 x Epson C70 Colour Inkjet Printer (not QL compatible)

3 x Epson 440 Inkjet printers

3 x Epson 500 Inkjet printers

2 x Canon BJ10 in cases and with paper feeder all working

1 x Canon BJC200 just needs Ink cartridges

1 x Kodak 200 printer, like a lexmark but with a compact flash reader. - no ink cartridges

1 x Lexmark Z11 - no ink cartridges

The printers will be cleaned and tested prior to shipping. We are offering the Epson 850s for £35 each plus £15 p&p (within the UK), without any ink cartridges (as the ink tends to leak during carriage). We will however, supply an unused set of black and colour Epson ink cartridges for the 850s for £10 extra at time of ordering.

QEMULATOR UPDATES

from Daniele Terdina

Q-emulator 2.3.3 for Windows is now available at <http://users.infoconex.com/daniele/winql.html>.

The update is free for registered users.

Changes since last version:

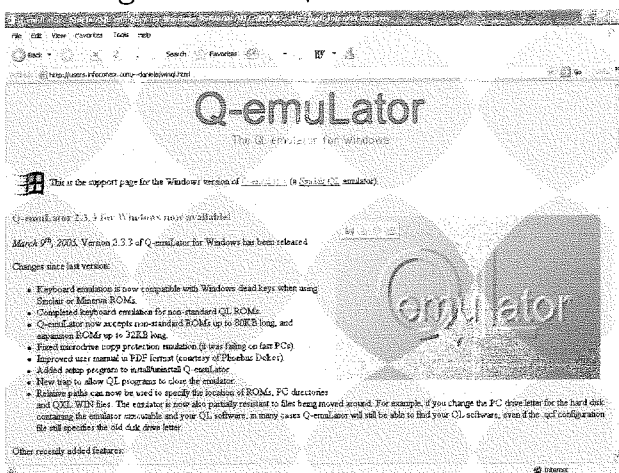
- Keyboard emulation is now compatible with Windows dead keys when using Sinclair or Minerva ROMs.
- Completed keyboard emulation for non-standard QL ROMs.
- Q-emulator now accepts non-standard ROMs up to 80KB long, and expansion ROMs up to 32KB long.
- Fixed microdrive copy protection emulation (it was failing on fast PCs).
- Improved user manual in PDF format (courtesy of Phoebus Dokos).
- Added setup program to install/uninstall Q-emulator.

A minor update to the Mac 68K version is also available, as Daniele was able to compile it again by using a Mac emulator for Windows. You can download it at

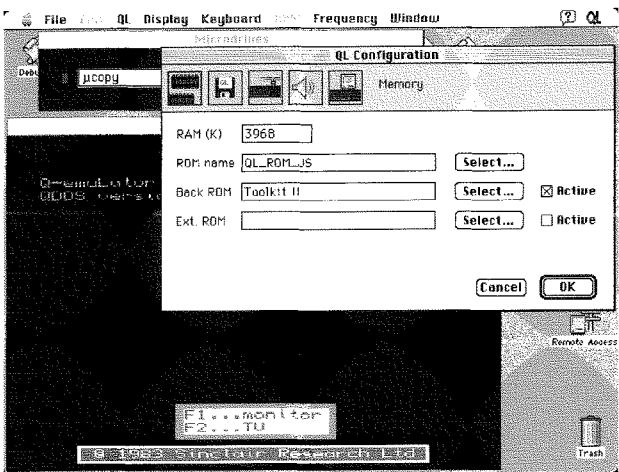
<http://users.infoconex.com/daniele/MacQL.html>

Changes since last version:

- Reads files created by the Windows version of Q-emuLator.
- Fixed microdrive copy protection emulation (it was failing on fast Macs).



QemuLator in Windows



QemuLator in Mac environment

QEMUFAST

News from Daniele Terdina

QemuFast is a new Sinclair QL emulator that runs many times faster than any other existing QL emulator. (Last time I checked, it was running 4x faster.)

Much of its code derives from Q-emuLator, except for the CPU emulation that has been completely rewritten.

A prototype has been sitting on my hard drive for a couple of years now, but it has never been merged with the standard Q-emuLator for a few reasons:

1) Speed comes to the price of compatibility. Although QemuFast can already run lots of QL software, compatibility with very old QL software is reduced compared to Q-emuLator and it can't be improved by much (without giving up the vast

speed advantage). Excellent compatibility with a real QL is one of the major goals of Q-emuLator. To be fast, QemuFast also trades off some extra checks that Q-emuLator does to insulate the user against misbehaving QL software. It is possible for a buggy QL program to crash the QemuFast application (but not Q-emuLator).

2) The rewrite has been a major effort, and resulted in a very visible speed increase. Coupled with the decreased compatibility, it would make more sense for it to be a separate product, with different goals.

3) QemuFast has not been tested extensively, yet.

4) It doesn't still include all of the CPU emulation optimizations that I had originally planned.

While QemuFast is still a prototype, it doesn't do anybody any good just by continuing to sit on my hard drive, so I've decided to release a first alpha-quality version that can only be used until the end of May 2005. It is available to all Q-emuLator users who purchased the "Expanded QL" level of registration and it can be downloaded by using this link:

<http://users.infoconex.com/daniele/win/QemuFast.zip>

If you don't have an "Expanded QL" Q-emuLator registration but you want to try QemuFast, you may ask me for a free temporary registration code.

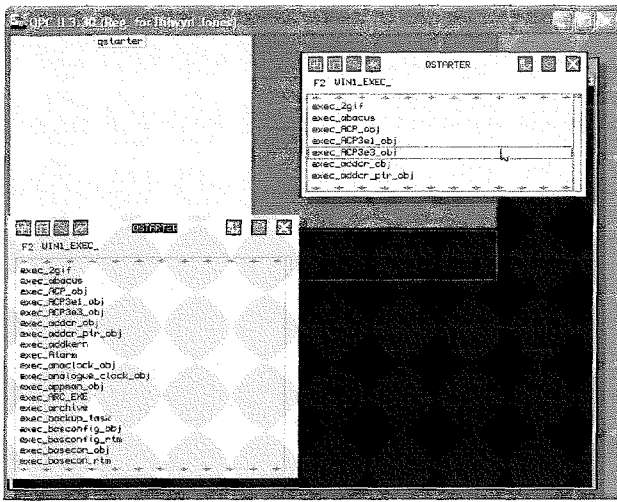
If you try QemuFast, please let me know of any problems you encounter running QL software with it.

QSTARTER

This is my first real venture into GD2 programming, thanks to the forthcoming Easyptr 4 update from Marcel Kilgus. This program features a choice of System Palette colour scheme or the traditional QL mode 4 scheme, with scalable menus for true resizing.

Although to some extent an experiment in GD2 programming for me, the program nonetheless has a serious use. It is a program launcher which searches your hard disk (or floppy disks if you wish) for executable program files and displays them in a menu for you to select which to execute. Keep it in a QPAC2 button frame if you wish for convenient program starting.

Download it from my usual Tesco.net website: <http://homepages.tesco.net/dilwyn.jones/software/freeware/freeware.html>



Qstarter, one copy running in mode 4 colours, the other in a grey System Palette theme although it may be hard to tell them apart in black and white!

QFILES

A little program from Dilwyn Jones which provides a button which activates the File_Select\$ menu from QMenu, and stuffs the selected filename into the stuffer buffer ready for retrieval with the usual ALT SPACE keypress.

This program provides a convenient method of selecting files via the pointer driven File_Select\$ menu, even if indirectly, for older non-pointer driven programs used in the pointer environment. With this little program you could select files from a menu instead of typing in filenames for older programs like Quill, for example.

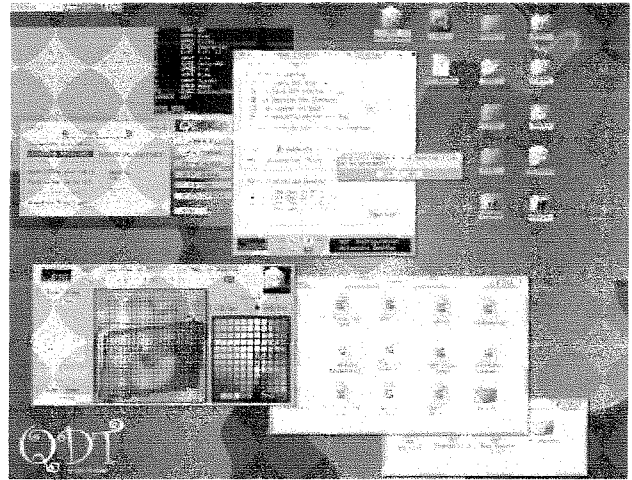
Download it from my usual Tesco.net website: <http://homepages.tesco.net/dilwyn.jones/software/freeware/freeware.html>

QDT

Jim Hunkins writes:

The next release of QDT is delayed due to my eminent move to Toronto, Canada to accept a new position as an architect for ATI. QDT work is proceeding but at a slow pace during this period but I would hope to have the next update out by the end of May. Expectations are that it will include the much desired drag and drop functionality, among many other improvements.

As soon as the move details are known, they will be posted on the QDT website and announced here. The QDT email addresses and website will remain intact during and after the move.



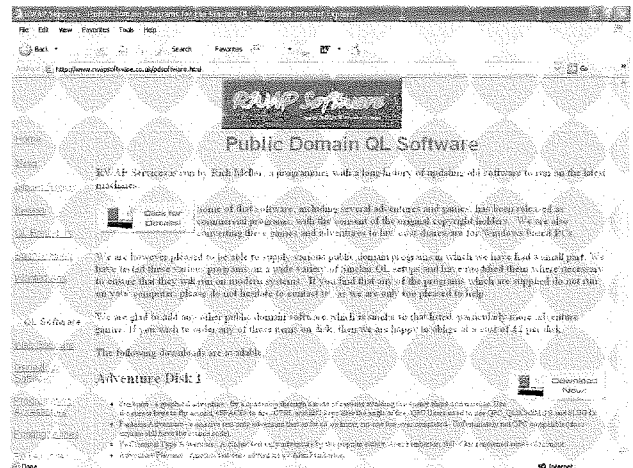
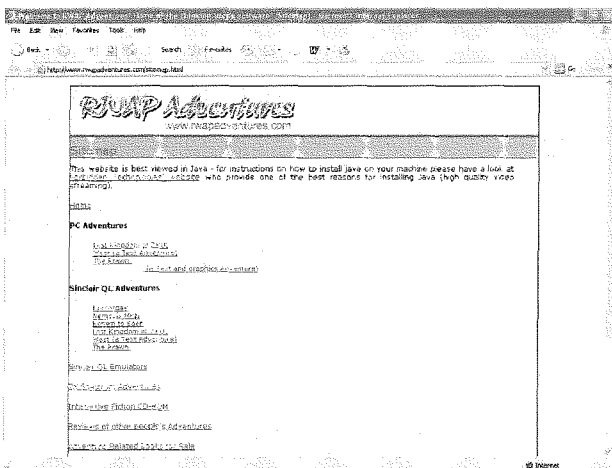
QL Software on the Web - Update

Rich Mellor has asked us to add details of his adventures, games and PD software web pages to the list of software sources that we published in recent issues.

These sites may be found by following links from

the main RWAP Services home page. The site includes both information about commercial QL adventures and games and freeware titles which may be downloaded. Visit Rich's home page at:

www.rwapsoftware.co.uk



Helpline

Question: I have set up an icon in Launchpad to start a program which needs a specific DATA_USE and PROG_USE setting. I've specified these in the program properties and when the program is started it all works fine. If I switch to another program, do some work to that, then go back to the first program, it seems a bit confused about where to look for its files.

Answer: Another program may be altering the DATA_USE or PROG_USE settings. There is only one set of DATA_USE and PROG_USE settings provided by Toolkit 2 or SMSQ/E for the whole computer, not a separate setting for each program as some people might prefer.

Even if Launchpad sets these correctly when your program starts, it has no control over other programs altering the setting. There is a facility in Launchpad to work around this and reset the settings for a given program. Right click on the icon for your program (i.e. DO or press the right mouse button while the pointer is over the icon) and select the Set And Pick command. This will "force" all the settings you specified for your program and jump back into your program if it's still running. Alternatively, pick the Set command just to restore the settings without picking your program directly. Another option is to go into the MySQL menu, select the System command and manually alter the relevant settings in there.

Question: I have some old protected QL software, but no longer have a QL. Can I still run these programs on any emulators or new QL systems?

Answer: [Daniele Terdina supplied the following information:] In fact, Q-emulator can run most copy-protected QL games (for example, the unpatched Psion Chess 1.0). But you need to own the original microdrive, have a QL capable of reading it, and a way of moving files between the QL to the PC (this usually means a QL with a floppy disk interface). Q-emulator includes a utility to copy the copy-protected files from microdrive to another location while preserving the copy protection information.

Questions? Write to Editor@QLtoday.com



JUST WORDS!

LET'S PARTY...

Parabéns!
Gratulerar!
Congratulazioni!
Yr act o longyfarch!
Tillykke!
Tebrikler!
συγχαρητήρια
恭喜 恭喜

10

Congratulations!
Félicitations!
Herzlichen Glückwunsch!
Gefeliciteerd!
Felicitaciones!
Go maire tú!
Gratulálok!
Поздравляю!
おめでとうございます

JUST WORDS! CONGRATULATES QL-TODAY ON ITS 10TH PUBLISHING YEAR.

Geoff Wicks, 56 Peveril Crescent, West Hallam, Derbyshire DE7 6ND, U.K.

Tel: +44 (0)115 - 930 3713

email: gwicks@beeb.net

Web: <http://members.lycos.co.uk/geoffwicks/justwords.htm>

Big Bangs and After-Glows

Geoff Wicks

By the time you are reading this the UK will probably be in the midst of a General Election campaign, so, in true politician's style, I shall start this article by rubbishing the opposition.

Bill Gate's once sharply criticised car manufacturers for their lack of imagination and innovation, and said they should follow the example of the computer world. One car manufacturer reminded him that Microsoft's great contribution to computing was a button marked "Start" that you used to turn off the computer. I have a better one than that. Every day I access a website giving the BBC's 5 day weather forecast for my nearest town. There is a mass of statistical information on this site, but the figures are too small to read. To make them bigger I have to click a button marked "smaller".

The Microsoft logical thinking virus has now crossed species and may have infected the QL community. This year in the QL world the After-Glow will occur before and not after the Big Bang.

Getting confused? Don't blame me! Blame Sjef v.d. Molengraaf! This year, Eindhoven's international show will be held on 18th June and Sjef thought up the name "QL2004 After-Glow Show". But this year it is the UK's turn to hold the "Big Bang" this will be Quanta's "QL is 21" show in Portsmouth on 29th and 30th October.

First Eindhoven. When Sjef suggested the title "QL2004 After-Glow Show" I felt my heart miss a beat, because the name expressed exactly what I would like the June Eindhoven show to be.

One of the great successes of QL2004 was that, for the first time in years, we got most of the QL's "movers and shakers" together for a hard working day and a relaxing social evening. This was good for the QL, because since QL2004 there has been a burst of development activity.

Eindhoven is a geographically convenient centre for QL developers to come together, and Sjef and I want to recreate something of the QL2004 atmosphere in June although on a smaller scale. Hence the name "After-Glow Show". We envisage the main developers coming together for a brainstorming session, demonstrations of some of the new developments for the "punters" and then a meal together in the New Canton.

Please send us your reactions to this idea, and suggestions for topics and activities. This will probably be the last QL-Today before the show itself, so watch out for developments via the QL-users email list.

The "Big Bang" this year will be the QL is 21 two day show in Portsmouth. It will be held in the Ibis hotel on 29th and 30th October. Quanta has a top notch team to run this show. Roy Brereton, who organised QL2000, is the general organiser, I am responsible for the activity programme and Sarah Gilpin, who has organised many Manchester shows, is responsible for the so-called "Ladies Programme".

Producing an activity programme for QL is 21 will not be easy. To understand why you need to know something about the UK QL scene and the membership of Quanta. The UK has always had the highest number of QL users in the world, and many are people who embraced the QL as a cheap and simple way into computing. Some have dabbled with simple programming, but for many the QL has been a tool to follow a specialist interest or hobby. They may not be programmers, but they are doing clever things with their QLs.

The recent survey of Quanta members and other QL users showed that just under a fifth of Quanta members are still using basic systems including microdrives. Extrapolating from this it is probably true to say that about a third of Quanta's members are not interested in the pointer environment, the new colours or any of the recent QL developments. At the same time probably about the same number of members are enthusiastic supporters of QL development.

At Portsmouth Quanta has to cater for all of its members and that means an activity programme of interest to both QL developers and black box/microdrive users. This is a tall order and is one of the reasons Quanta is carrying out a consultation exercise to help us plan QL is 21.

The first stage of this consultation is an informal meeting at the Hove Show, open to both Quanta members and non-members, to discuss what QL-users would like to see at Portsmouth. If you cannot make it to Hove, then your opinions are welcome by email, snailmail or telephone. My contact details can be found in the Just Words! advertisement.

At Portsmouth I am throwing out a big challenge to QL developers. Over the last few years and months the QL has developed at an amazing pace with more colours, better sprites, graphical user interfaces, faster machines, new sound possibilities, improved printer access and finally the holy grail of internet access. How well can you explain to a black box user what we are doing, and why these things are important for practically every QL user and for the survival of the QL?

Gee Graphics! (on the QL?) - Part 42

H. L. Schaaf

Hyper G Graphics?

Distracted again and trying to 'procrastinate later'. Thanks for the kind 'postcard' comments in the QL user mail. I also lurk in usenet newsgroups and noticed that Mathematica has a location (integrals.wolfram.com) where you could have them integrate mathematical expressions and get almost instant answers. So I tried it with the expression we've been using for Lamé curves. It came back with a "hypergeometric equivalent". What's that and how do I make use of it? More research is needed.

As you know the web is just the tip of the internet and in addition to Google I've looked at vivisimo and incy wincy as search engines to see if anyone has the length and area of compound Lamé curves worked out. No luck so far.

Hypergeometrics is another of those things that the great names of mathematics, Wallis (1655), Euler (1769), and especially Gauss (1812) knew about long ago and the subject has enjoyed renewed interest and development with computers in the hands of the Borwein brothers and many others. So a little challenge to myself is to

teach the QL how to do hypergeometric series and functions.

A beginner's page at www.webpearls.com/hypergeo is a good starting point. Enjoy.

The hypergeometric approach gets us the area under Lamé curves and the answers agree with the Allard and Grule methods, to at least 5 digits, over most of the range. When we have a choice of 3 different answers how are we to choose? We are still looking for the length. Why? Just for Fun.

The listing GHS_bas takes Lamé parameters and puts them into the Gaussian hypergeometric series and gets us the area under the curve in the first quadrant. It is set to show up to 10 by 10, but the QL will happily compute for off-the-screen values. The dot maker at line 850 lets us know it is working and when done it prints out the number of terms that were used. The eval FuNction at line 1020 lets us put in simple expressions having no more than one operator. When I get back to it I'll add SQRT, LN, EXP trig functions, etc. That's what I said about the VAL FuNction in GG#32, early in 2003. There must be a parser ready to use somewhere in the vast QL software repository. Anybody know?

Next time? Maybe a new slant on getting the arc length.

```
100 REMark GHS_bas
110 REMark HL Schaaf Mar 22, 2005
120 REMark for GG#42
130 :
140 WMON
150 PAPER 0 : INK 7 :CLS
160 PAPER #2,0 :INK #2,7 : CLS#0
170 :
180 eps = 2-31
190 :
200 get_Lame
210 makegrid
220 show_Lame
230 CLS#2
240 hyperLame
250 quityet
260 :
270 DEFine PROCedure quityet
280 PRINT #0;\ '[R]epeat, [A]nother or [Q]uit ?'
290 REPeat again
300 ans$ = INKEY$(-1)
310 IF ans$ == 'r': GO TO 210
320 IF ans$ == 'a': GO TO 200
330 IF ans$ == 'q': STOP
340 END REPeat again
350 END DEFine quityet
```

```

360 :
370 DEFine PROCedure show_Lame
380 AT 0,5:PRINT '(X/';lama;')^';lamn;
390 PRINT ' + (Y/';lamb;')^';lamm;' = 1'
400 INK 2
410 FOR x = 0 TO lama STEP lama/32
420 POINT x,f(x)
430 END FOR x
440 INK 4
450 FOR y = 0 TO lamb STEP lamb/32
460 POINT invf(y),y
470 END FOR y
480 INK 7
490 END DEFine show_Lame
500 :
510 DEFine FuNction f(x)
520 REMark given x, find y
530 RETurn lamb*((ABS(1-((ABS(x)/lama))^lamn))^(1/lamm))
540 END DEFine
550 :
560 DEFine FuNction invf(y)
570 REMark given y, find x
580 RETurn lama*((ABS(1-((ABS(y)/lamb))^lamm))^(1/lamn))
590 END DEFine
600 :
610 DEFine PROCedure makegrid
620 LOCAL I
630 PAPER 0: INK 7
640 SCALE 12,-.75,-1:CLS
650 LINE 0,10 TO 0,0 TO 10,0
660 FOR I = 0 TO 10
670 CURSOR I,0,-4,4:PRINT I
680 LINE I,0 TO I,-.25
690 CURSOR 0,I,-10,-4:PRINT I
700 LINE 0,I TO -.25,I
710 END FOR I
720 END DEFine makegrid
730 :
740 DEFine FuNction GHS(a_h,b_h,c_h,z_h)
750 sum_h = 1 : add_h = 1 : count = 0
770 REPEAT series
780 num_h = (a_h+count)*(b_h+count)*z_h
790 den_h = (c_h+count)*(count+1)
800 add_h = add_h*num_h/den_h
810 sum_h = sum_h + add_h
820 IF ABS(add_h) < eps : EXIT series
840 count = count + 1
850 PRINT #0;'. ';
860 END REPEAT series
870 PRINT #0;\count;' terms'
880 RETurn sum_h
890 END DEFine
900 :
910 DEFine PROCedure hyperLame
920 g_hs = GHS(a_h,b_h,c_h,z_h)
930 show_GHS
940 IF neqm THEN
950 area = AxB*((xeqy*2*g_hs)-(xeqy*xeqy))
960 ELSE
970 area = AxB*z_h*g_hs
980 END IF
990 AT 1,16: PRINT 'area = ';area
1000 END DEFine hyperLame
1010 :

```



```

1020 DEFine FuNction eval(n_$)
1030 opt = 'PI' INSTR n_$
1040 IF opt THEN
1050 n_$ = n_$(TO opt-1)&'3.1415926535'&n_$(opt+2 TO)
1060 END IF
1070 n_ = n_$
1080 opt = '/' INSTR n_$
1090 IF opt : n_ = n_$( 1 TO opt-1) / n_$(opt+1 TO)
1100 opt = '*'INSTR n_$
1110 IF opt : n_ = n_$(1 TO opt-1) * n_$(opt+1 TO)
1120 opt = '+'INSTR n_$
1130 IF opt : n_ = n_$(1 TO opt-1) + n_$(opt+1 TO)
1140 opt = '-'INSTR n_$
1150 IF opt>1 : n_ = n_$(1 TO opt-1) - n_$(opt+1 TO)
1160 RETurn n_
1170 END DEFine
1180 :
1190 DEFine PROCedure get_Lame
1200 neqm = 0 : area = 0
1210 INPUT ' Lame_e1 (horizontal exponent) ' ;ln$
1220 lamn = eval(ln$)
1230 INPUT ' Lame_e2 (vertical exponent) ' ;lm$
1240 lamm = eval(lm$)
1250 INPUT ' Lame_a (horizontal half-axis) ';la$
1260 lama = eval(la$)
1270 INPUT ' Lame_b (vertical half-axis) ';lb$
1280 lamb = eval(lb$)
1290 a_h = 1/lamn
1300 b_h = -1/lamm
1310 c_h = 1+(1/lamn)
1320 z_h = 1
1330 AxB = lama*lamb
1340 REMark for the symmetrical case
1350 IF lamn == lamm THEN
1360 neqm = 1
1370 REMark x=y for midpoint
1380 xeqy = (.5)^(1/lamn)
1390 z_h = xeqy^lamn
1400 END IF
1410 END DEFine get_Lame
1420 :
1430 DEFine PROCedure show_GHS
1440 PRINT #2;" Gaussian Hypergeometric Series"
1450 PRINT #2;' 2F1( ' ;a_h;' ;b_h;' ;c_h;' ;z_h;' )'
1460 PRINT #2,,'= ' ;g_hs
1470 END DEFine show_GHS
1480 REMark end of listing GHS_bas

```

Letter-Box

Bruno Coativy writes:

I would like to thank Norman Dunbar for his excellent article "Recursion in Assembly Language" (QL-Today, Volume 9, Issue 5, Page 46). Nevertheless, I disagree with him as to the use of the LINK/UNLK (see Page 50) instructions.

Norman has not taken into account the movable nature of the SuperBasic interpreter that makes this use particularly unsafe. This subject has already been covered in QUANTA Magazine Volume 19, Issue 10, Page 12 and Volume 19, Issue 11/12, Page 5. See also the QL Technical Guide or the QDOS/SMS Reference Manual: 9.6 Code Restrictions.

Another limitation to using recursive programming in assembly language lies in the fact that the stack of the SuperBasic JOB has a quite limited size (128 bytes?). While QDOS knows how to adapt this size to the interpreter's needs, it

seems there is no known way of doing the same thing from within a CALLED code.

So, in my opinion, the only way to run a deeply recursive programme in assembly language under QDOS is to run it from another JOB than SuperBasic. In this case, you can set up a huge stack, memory permitting. And this time you can use the LINK/UNLK instructions, because this JOB is not likely to move in memory.

Regarding the direct question in the letter (updates on QDOS/SMS Reference Manual): there is a small pile of collected changes so far, but the time situation...

Jim Hunkins, author of QDT, responds to Per Wittes review in the previous issue:

I appreciate Per taking his time to work with QDT and put together his review. I found his approach to the review unique and well done, considering the other information previously published or on the web, as he pointed out.

I would like to clarify and/or respond to a few of the points that Per brought up.

On the light side, a minor clarification as to my life at Apple Computer for the last 5 years, I was actually there in the roll of technical lead and senior hardware engineer, not as a software engineer (would have been fun). However, I have been working with software as Per pointed out periodically for most of my career and was involved with Apple software to different degrees on the system and graphics level during my stay there.

As to the comment that living in sunny California should be such a privilege that I should give away QDT, I should actually charge a lot more because working on it keeps me from going out there and enjoying the sunshine like everyone else!

Now for some serious items...

The QDT log file is currently left open while running QDT. For most users, this is transparent and works well. I was surprised by what Per is attempting (running multiple copies of QPC2 of the same disk set) that this causes trouble with but QL users are the creative type. Some of the reasons for keeping the log file open are no longer critical so it should be possible to close the file when it isn't in use. There are some potential side effects to this but I will consider this change for an upcoming release. Regardless of this change, the user should be aware that only one copy of QDT can be running at a time, due to the file and memory sharing and message passing that it does (some of the tricks that I do to put PE

through its paces). Since QDT is a desktop management program, this should be a reasonable restriction.

Referring to the jobs that QDT uses, a minor correction is that there is only one job per desktop object. Only if an object is executed or opened to an open folder is a second job started. Normally, I don't recommend cluttering the desktop with too many objects, keeping it free for doing real work on. The user should instead take advantage of QDT being designed to use folders extensively. QDT supports folders within folders. Objects within folders do not have separate jobs running, only objects on the desktop itself.

Currently I would agree that it is confusing when you look in your job tree and see the jobs that QDT adds, especially with the same names. A future rev will use a new job naming convention to help the user, if needed, identify which job is associated with which object/folder. The other thing on the roadmap that will help is that QDT will eventually have its own job management utility. This utility will allow the user to filter out QDT, system and other types of jobs, along with including other useful functions. Both OS X and Windows has similar problems with often 100s of jobs running, mostly for system level work that the user doesn't care about. I would consider most QDT jobs to fall into this type of category so the filtering them out in the utility matches exactly what these other OS's do.

Looking at the number of jobs, early testing during the development of the architecture for QDT showed that system overhead for this situation was virtually non-existent in the pointer environment. Most of the QDT jobs are actually held in a wait state (suspended) until messages or mouse clicks get passed to them.

There is an overhead to loading the desktop for the first time due to the jobs which I am not happy with, but it is part of SMSQ/E, not QDT. Many users avoid this by just keeping QDT running and are therefore not affected by this on a day-by-day basis. I understand that Roy doesn't close QDT down and has no problems with it kept open. QDT has gone through quite a bit of debugging and work to make sure that there are no memory escapes, etc. that plague some programs if left running too long).

QDT does take up memory as it runs. My personal running version right now is using just about 1 M of memory. Since most QDT compatible systems do have 4+ M of memory, this is not that bad. It should be noted that, while not everything is yet functional in QDT, its core is a substantially large and complex program, what

with all the graphics and new functions put on top of the pointer environment. There isn't much that can be done to get around this. I do have on my list to do more shared code and other optimizations but the priorities are in favor of new functionality before reducing the memory footprint, since this footprint should not have a negative impact for most QDT users.

My last note for now, Per stated a very true fact that for a new user with 'anything less than an Aurora', setting up the environment for QDT would be very difficult. This is true especially since QDT requires an Aurora or other equivalent capabilities to support the graphics (it will not run with anything less capable than the Aurora). In reality, if a user sets the minimal PE setup in their boot file, then the QDT setup does take care of everything else. It even tells the user what to add to his/her boot file if it finds something missing and can't handle it automatically.

Again, I want to express my appreciation to those who have taken their time to test and write about QDT. All this input will hopefully be useful for the various readers and I do carefully review all comments to look for things that I can either clarify or add to the feature list of QDT.

I am looking forward to hearing more feedback and requests.

Richard W. Parker, Jr. writes:

QL Today Publishers, Jochen, Dilwyn and Bruce, and all your helpers, this last issue was a great one. QLT is my life line to the QL world and I appreciate your efforts to keep us QLing and up to date, immensely. The news on QPCPrint was welcome and longed for, as my QPC2 is just a toy now. Thank goodness my old QL still runs well, but its only output is the old 9-pin Sinclair printer.

Roy Wood's 'Essential Information' is a needed series, once in a while, to keep all the available QL stuff in context and bring us 'outbackers' into the current QL world. I'll probably reread that series several times.

I'll miss the 6th issue but have no problem with your reasoning for just doing 5. Going into a 10th year is something to be very proud about, especially considering the shorter life span of your glossy predecessors. I agree that you don't need the frustration and worry of trying to pull the summer issue out of a hat.

Congratulation and thanks.

... and thank you all for writing to us - we have not had a letter-box section for quite a while. And wow, positive, encouraging feedback ... more of that, please ; -)

Scaling in EasyPTR 4

Dilwyn Jones

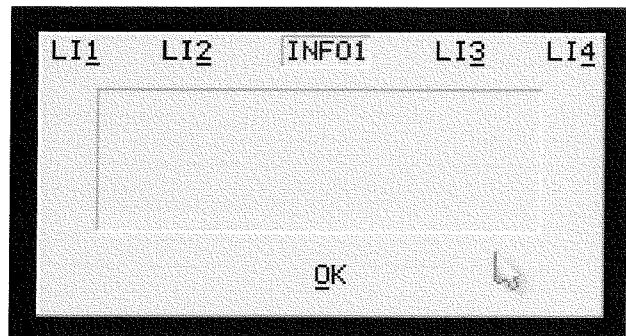
Marcel Kilgus has put in a lot of work on Easyptr to get object scaling flags working (among other things!) so that we can create scalable menus with Easyptr. I've been learning to use it and here's my findings to help anyone struggling with it, or who'd like to use it but maybe doesn't have the time to learn by playing with it.

Obviously, feedback, corrections, improvements etc welcome. This is based on a few days of playing and little documentation and examples, so may not be wholly correct, or there may be ways of improving the examples given and I don't mind being told that!

In older versions of Easyptr the only standard way of resizing menus was to create multiple definitions. If your computer has QL, EGA, VGA and SVGA screen modes available, you had to create four different sized versions of the same menu and include all four menus in your program and write code to select the appropriate definition for whatever screen resolution you used at the time. You can still do that, and programs which work like that will still work, but

there is now an extra method which can work rather better and is certainly more flexible in terms of sizes - you can scale menus by specifying a minimum size and how objects such as loose items, information windows and application windows are resized when the outline itself is resized.

I created a menu with 4 loose items and a title information window across the top, an application window in the centre of the menu and finally one loose item at the bottom. Why choose this layout will hopefully become clear when we start scaling elements.



(ASCII diagram on the next page)

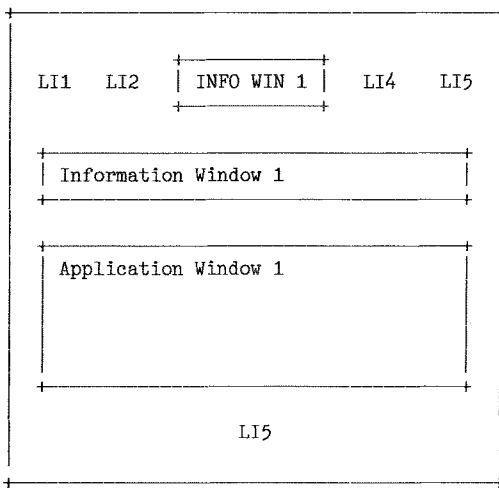


Figure 1: Original size menu

Loose items 1 to 4 represent a pretty normal row of loose items across the top of the menu - often they'd be Move, Resize, Sleep, and ESC for example. Information Window 1 is centred across the top as a title, LI2 is about a quarter of the way across and LI3 is about 3/4 of the way across. LI1 is near the left edge, and LI4 is near the right hand side of the menu.

This positioning of the items across the top is pretty important for the explanation of the different scaling values. When the menu is resized, they do not need to move down, but they do need to move across in such a way that loose item 4 needs to end up towards the right of the menu after resizing, information window 1 needs to remain pretty centred and LI2 and LI3 need to end up about a quarter and three quarter the way across. The loose items should not change size. The title window can either stay the same size or be scaled, either way it should stay centred in the same vertical position.

Application window 1 will be resized to take advantage of an enlarged menu to display more of the application window menu (e.g. a list of files). Its origin (top left) stays where it is.

LI5 is not to be resized (it's an 'OK' loose item) but it will need to be moved when the menu is resized otherwise it'll get hidden by the enlarged application window.

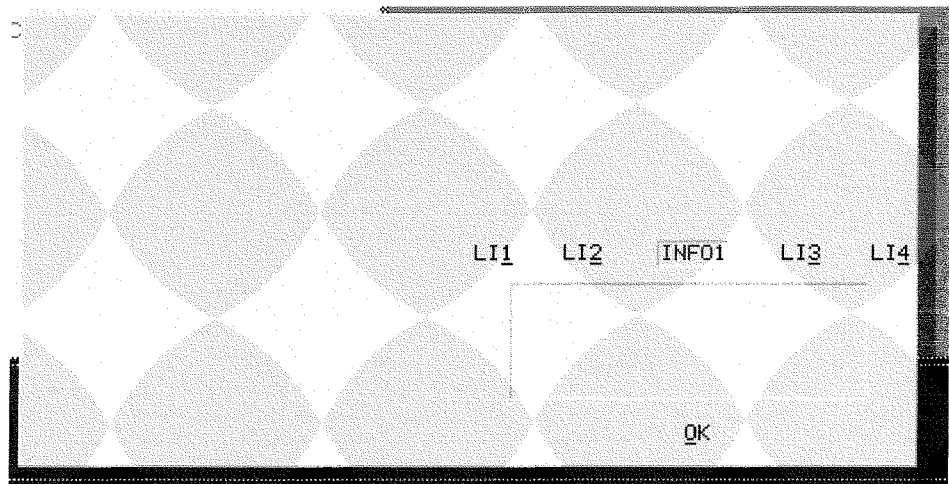
We'll start off with a menu 200 pixels across and 100 pixels down. This is referred to as the 'minimum size'. It will never get smaller than this and all resizing is done with reference to this size.

The actual size of 200x100 is, strictly speaking, the size of the area inside the border, i.e. it does NOT include any border around the menu or any shadow under and to the right of the menu. So if you have a border width of 1 and shadow depth of 2, a menu of size 512x256 will not fit in a 512x256 QL display! I've used 200x100 simply to use nice round figures to make calculations of changed sizes easier!

Figure 1 above shows the original size of menu, the smallest size it can go.

Before we go any further, I should note that scaling works with reference to the change in width and height of the menu. If you think about it, it's just about the only sensible way of doing things! If a menu changes from 200 pixels wide to 400 pixels wide, the horizontal scaling is best thought of as a change of 200 pixels, not necessarily as doubling or any percentage as such. Within Easymenu we set scaling flags which will cause size or origin to be changed by 1:4, 1:2, 3:4 or 1:1 of the change in width in pixels. So if width changes by 200 pixels, anything with a scale flag of 1:4 would change by a quarter of that 200 pixel size change. Anything with a scale flag of 1:2 would change by half that 200 pixel size change. Anything with a scaling flag of 1:1 would change by the full 200 pixels size change. The relevance of the four possible scale values will hopefully become clear when we discuss positions.

First of all, we'll scale the origins for everything 1:1 and see what happens when we double the menu size. This alters the width by 200 pixels across and 100 pixels down, so 1:1 scaling adds 200 pixels to the origin of everything across, 100 pixels to the origin of everything down. The results are in figure 2.



TF Services

Compswitch

A UK 4-way trailing socket designed to switch off computer peripherals automatically when the computer is switched off, or (in the case of an ATX computer) when it auto-powers down. *Compswitch* has one control socket, and three switched sockets. Can be used with lights/hifi/monitors—ie a QL monitor can be used as a switch control.

Cost **£24**

superHermes

A major hardware upgrade for the QL
All Hermes features (working ser1/2 at 19200, independent baud rates/de-bounced keyboard/keyclick) IBM AT kbd I/F // HIGH SPEED RS232 at 57600// serial mouse port and 2 other RS232 inputs// 3 I/O lines // EEPROM

Cost (including manual/software)£90 (£92/£93)
IBM AT UK layout Keyboard..... £11 (£13/£15)
Serial mouse..... £8 (£8.50/£9)
Capslock/scrolllock LED £1 (£1.50/£1.50)
Keyboard or mouse lead £3 (£3.50/£3.50)
High speed serial (ser3) lead..... £4 (£4.50/£4.50)

Hermes available for £25 (£26/£27) Working ser1/2 and independent input, debounced keyboard.

SuperHermes LITE: All Hermes features (see above) + an IBM AT keyboard interface only.
Cost (incl keyboard lead) £53 (£54/£55)

QL REPAIRS (UK only)

Fixed price for unmodified QLs, excl microdrives. QLs tested with Thorn-EMI rig and ROM software.

£27 incl 6 month guarantee

Minerva

The ORIGINAL system operating system upgrade

OTHER FEATURES COMMON TO ALL VERSIONS

DEBUGGED operating system/ autoboot on reset of power failure/ Multiple Basic/ faster scheduler- graphics (within 10% of lightning) - string handling/ WHEN ERROR/ 2nd screen/ TRACE/ non-English keyboard drivers/ "warm" fast reset. V1.97 with split OUTPUT baud rates (+ Hermes) & built in Multibasic.

First upgrade free. Otherwise send £3 (+£5 for manual if reqd).
Send disk plus SAE or two IRCs

MKI...£40 (£41/£43) MKII...£65 (£66/£67)

MINERVA RTC (MKII) + battery for 256 bytes ram.
CRASHPROOF clock & I²C bus for interfacing. Can autoboot from battery backed ram. Quick start-up.

QL RomDisq

Up to 8 mbyte of flash memory for the QL
A small plug in circuit for the QL's ROM port (or Aurora) giving 2, 4 or 8 mbytes of permanent storage - it can be thought of as a portable hard disk on a card, and reads at some 2 mbytes per second.
Think of it - you could fully boot an expanded QL, including all drivers/SMSQ etc off RomDisq at hard disk speed with only a memory expansion needed.

2 mbytes RomDisq.....£39 (£40/£41)
4mbytes RomDisq.....£65 (£66/£67)
8 mbytes RomDisq.....£98 (£99/£100)
Aurora adaptor.....£3 (£3.50/£4)

MPLANE

A low profile powered backplane with ROM port

A three expansion backplane with ROM port included for RomDisq etc. Aurora can be fitted in notebook case and powered off single 5V rail - contact QBranch for details. Two boards (eg Aurora and Gold Card/Super Gold Card/Goldfire fixed to base. Suitable for Aurora (ROM accessible from outside) & QL motherboard in tower case. Specify ROM facing IN towards boards, or OUT towards back of case.

Cost£34 (£35/£36)

I2C INTERFACES

Connects to Minerva MKII and any Philips I²C bus

Power Driver Interface 16 I/O lines with 12 of these used to control 8 current carrying outputs (source and sink capable)

2 amp (for 8 relays, small motors).....£40 (£43/£44)
4 amp total (for motors etc).....£45 (£48/£50)

Relays (8 3a 12v 2-way mains relays (needs 2a power driver).....£25 (£28/£29)

Parallel Interface Gives 16 input/output lines. Can be used wherever logic signals are required.....£25 (£27/£28)

Analogue Interface Gives eight 8 bit analogue to digital inputs (ADC) and two 8 bit digital to analogue outputs (DAC). Used for temp measurements, sound sampling (to 5 KHz), x/y plotting.....£30 (£31/£32)

Temp probe (-40°C to +125°C).....£10 (£10.50/£11)

Connector for four temp probes.....£10 (£10.50/£11)

Data sheets.....£2 (£2.50/£3)

Control software & manual (for all I/F).....£2 (£2.50/£3)

QL SPARES

Keyboard membrane no longer on sale
1377 PAL £3 (£3.50/£4)
Circuit diagrams..... £3 (£3.50/£4)
68008 cpu or 8049 IPC..... £8 (£8.50/£9)
8301/8302 or JM ROM or serial lead £10 (£10.50/£11)
Power supply (sea mail overseas)..... £12 (£19/£23)
Other components (sockets etc) also available

Prices include postage and packing (Airmail where applicable) Prices are: UK (Europe /Rest of world). Payment by cheque drawn on bank with UK address/postal order or CASH! I can no longer accept card payments as UK only does PDQ transaction. SAE or IRC for full list and details 22 Feb 03

29 Longfield Road, TRING, Herts, HP23 4DG

Tel: +44 (0) 1442-828254

Fax/BBS: +44 (0) 1442-828255

tony@firshman.co.uk

http://www.firshman.co.uk

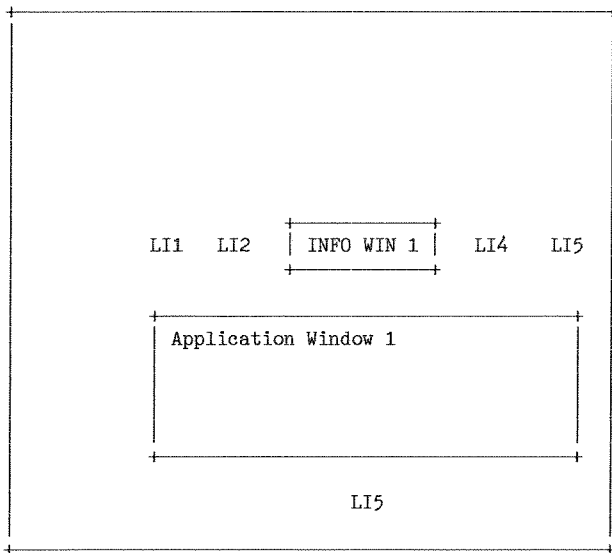


Figure 2: Doubling outline width and height and scaling x and y origins of menu elements

Oh dear. Not a pretty sight. Not quite what we had in mind.

Everything just got shoved over down and to the right.

It's not that simple after all.

Some of the items have to have the same origins and need to change size, others just need to be repositioned, so let's look on a case by case basis what's needed for each item.

If we scale both the origins and sizes, we'll run into even bigger problems - in addition to being bottom right of the menu, anything resized might well be too big to fit on the menu and may cause an error.

All loose items should stay the same size when the menu size changes, we just need to decide if, where and how to move them. So we set scale flags for width and height to 0 (no change in width or height when the menu is resized).

LI1 should stay in the same position (top left). So we can set its x and y origin scale flags to 0.

LI2 should be about a quarter of the way across the resized menu. If we scaled its origin 1:1 it would move 200 pixels to the right (200 pixels is the horizontal change of size) meaning it would end up 250 pixels across the menu. Too far. A quarter of the way across the resized 400 pixel wide menu is about 100 pixels across. It used to be about 50 pixels across. We will want it to be about 100 pixels across (a quarter way across the new menu size). 50/200 pixels is 1/4, so we use 1:4 scaling for the origin of LI2. This is scale flag value 1.

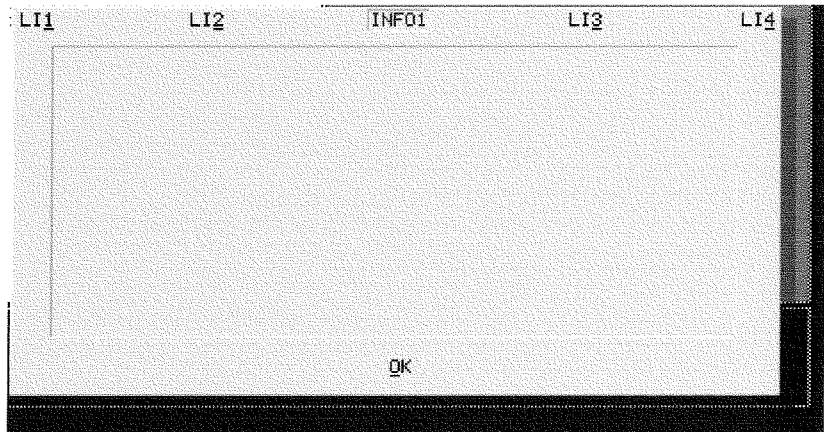
LI3 should end up 3/4 of the way across. It's about 150 pixels across the original menu, we'll need it to be about 300 pixels across the double-sized menu. So it needs to move by about 150 out of the 200 pixel increase in menu width, or about 3:4 in ratio terms. So we set a scale flag value of 3 for a 3:4 scaling.

LI4 should end up at the right hand end of the menu. So it needs to move by around the same number of pixels as the change in width of the menu. So we scale its origin by 1:1.

If you are mathematically inclined, you'll find this fairly easy, moving things by a proportion of the change in size to maintain the relative positions. It's a bit hard to visualise at first, so what I do is take example menu sizes and write down the old and new positions and sizes in pixel values to help me determine the correct scaling flags to use. A bit of trial and error and a few ugly menus later and you soon get the hang of it.

LI5 needs to end up at the bottom of the menu and halfway across. So it needs to move down by the same number of pixels as the increase in menu height (1:1) and needs to remain centred, so for every pixel increase in width it needs to move across by half a pixel. It was about 100 pixels across a 200 pixel wide menu originally, and in the new menu will need to be about 200 pixels across a 400 pixel menu, an offset of 100 pixels across (half the change in width). So we give it an x scale flag value of 2 (1:2) and y scale value of 4 (1:1).

The application window needs to grow to fit the space available, but as we aren't changing the size of the top row of loose items and title info window, its origin can stay in the same place. So we give both x and y origin a scale flag value of 0 each. Width and height of the application window can grow by the same amount as the change in menu size. So the scaling flag value for both width and height is set at 4 (1:1) for the application window.



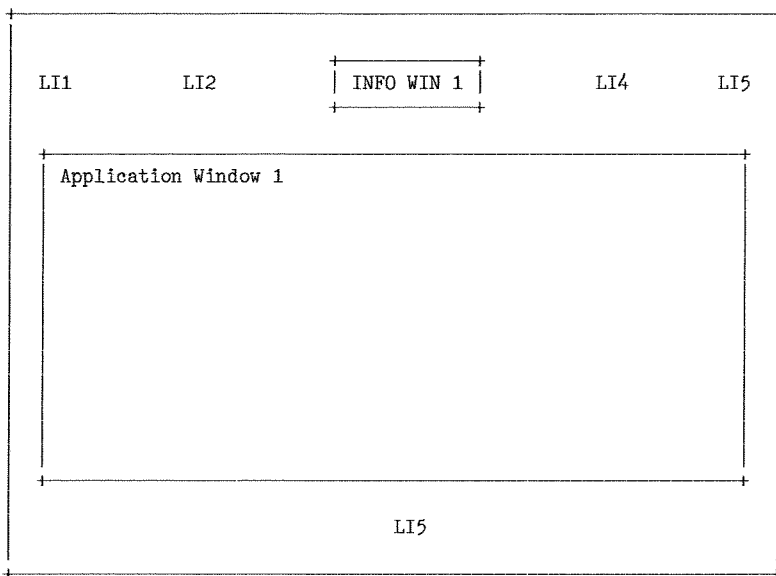


Figure 3: the desired perfectly proportioned result!

Figure 3 shows the new menu size and layout I had in mind. It's only an example - in practice you'll obviously come up with something tidier for your programs, hopefully this illustrates the points I was trying to make that:

- (a) which scaling flag value is used for an element depends on what end results you want to achieve, and
- (b) each and every element in a menu can have its own set of 4 scaling flags and all need not be the same!
- (c) sometimes, only origins need change, sometimes only width and/or height need change.

In the example above, the title window is simply repositioned centrally in the resized menu. It might look better if it grew with the menu, especially if your colour scheme includes a different title colour across the top of the menu. Here's one suggestion for how to achieve this.

Simply make the origin stay where it was and make the title grow in proportion to the change in menu width! If the title window is 40 pixels wide in the smallest menu, that leaves 160 pixels outside it or 80 on each side. In other words, its origin is 80 pixels across the menu. If it stays 80 pixels from the left when the menu grows from 200 pixels wide to 400 pixels wide, we simply add the full 200 pixels size change to it. In other words, we set the origin scaling to flag value 0 for x and y, height does not change so that also gets a scale flag value of 0, but the width changes 1:1 with the change in menu width.

So it gets a scale flag value of 4 for the info window width.

Thus you can see that there are two ways to maintain centering of an item:

- (a) if its width does not change, set width scale flag to 0 and x origin scale flag to 2 (1:2 scaling of origin)
- (b) if its origin is to stay in the same place, make its width grow 1:1 (width scale flag value 4) with the change in width of the lmenu itself.

The same principle applies to vertical positioning/centering if your program uses vertical columns of loose items rather than the usual horizontal rows, although this is obviously less common.

Obviously, how you approach scaling is ultimately up to you within the context of what you want to achieve with your menus. The things to be careful with are:

1. It is normally not a good idea to scale both origin and size 1:1 at the same time. Something may fall out of range, whether a window becomes too wide and too far over to the right or too far down to fit in your menu, or even fit on the screen!
2. When resized, will any resized element accidentally cover something else? In the example above, if the application window grows taller but loose item 5 stays in the same position, both will overlap and loose item 5 may be hidden, so we had to scale its origin vertically to allow for that as well as scaling horizontally to maintain centering. Equally, if we resize the title window across the top, be careful that the loose items to the right of it don't get hidden by it!

Setting Scale Flags In Easymenu

Assuming you have a version of Easymenu which permits the use of the scale flags (v4.01 or later, version number is shown at the bottom of the main Easymenu display, or when configuring Easymenu with Config or MenuConfig programs), here is how to set them in Easymenu.

Simply enter the list display for the element in question. From Easymenu's main menu, select the "Loose item" command, then right click or left click (depending on how configured) on the Change command to bring up the list display as shown in Figure 4.

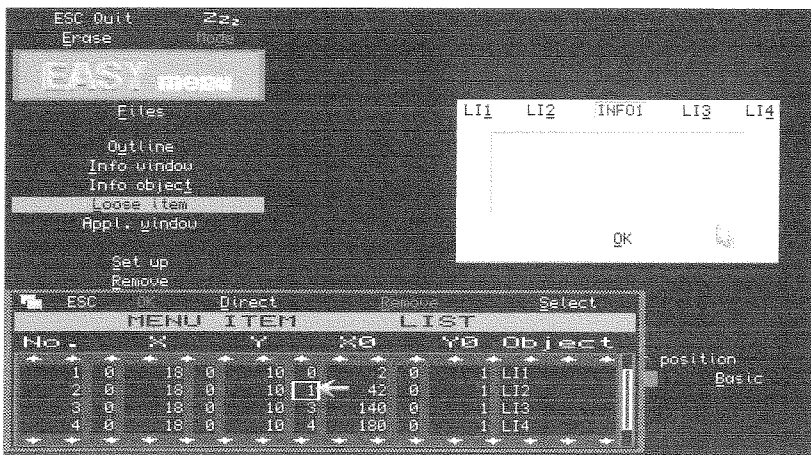


Figure 4: Menu item List, showing where scale flag values are entered.

This is the same as in older versions of Easy-menu, but the previously unusable narrow boxes to the left of each of the 4 columns of values you could enter values into are now usable. There is only room for one digit, which must be a value from 0 to 4 inclusive. To enter them, you simply click in the box for the entry you wish to change. Some people never use this screen - they simply design menus by dragging the corners etc in the visual editing of menus. To set scaling flag values, you need to know how to use this screen.

The numbers in the column under the heading "No." refer to the item number, in this example we see the first four loose items. The values under X are the loose item widths, the ones under Y are the loose item heights, and X0 & Y0 are the origin co-ordinates within the menu. The "Object" column simply shows the first few characters of a text object, or spr. if the object is a sprite, for example. If trying to reach an information object, you'd have to select the information window number by clicking on the item number in the left hand column then click on the Select command at the top to reach the list of info objects for that info window (an info window can have more than one info object of course).

For large, complex menus entering scaling flags in this way can be very tedious, since there are 4 flag values for each and every item in the menu. My only answers to this are:

1. Flag values default to 0, no scaling, which is often the value of the majority of flags in a menu.
2. Which is more tedious, entering them here or designing multiple versions of each menu in different sizes, as you used to have to before Marcel implemented scaling flags?

The scaling flag values possible are as follows:

- 0 = no scaling (no change in value when menu is resized)
- 1 = scale this element by 1:4 with reference to menu size change
- 2 = scale this element by 1:2 with reference to menu size change
- 3 = scale this element by 3:4 with reference to menu size change
- 4 = scale this element by 1:1 with reference to menu size change

Writing A Program To Use Scaling Menus

All this theory is great (light bedtime reading?!?), how do we write a BASIC program to make use of the scaling flags?

Specifying a new size is quite easy, using an existing version of the MDRAW command to set up and draw the menu. The full syntax of this command is:

```
MDRAW #ch%, menuname, xpos%, ypos%, xsize%, ysize%
```

#ch% is the basic window channel number in which the menu is to be drawn.

"menuname" is the name or filename or even address of the menu definition. If loaded from disk, it should be a full filename such as "ram1_mymenu_men", or the simple menu name if appended to one of the Easyptr extensions files ("mymenu"), or the address of the menu definition can also be used if required.

xpos% and ypos% are the origin co-ordinates of the menu. Usually, this will be allowed to set its own existing value rather than forcing it to a given part of the screen. In our examples, we'll set these to -1 to indicate no change of position.

xsize%, ysize% are the new width and height respectively of the menu.

If all you want to do is to make the menu as large as possible to fit the existing screen mode, check the screen width and height with SCR_XLIM and SCR_YLIM extensions in SBASIC (or use my DISPLAY_CDE extensions from a previous QL Today if you want your program to work on QDOS as well) and pass the new size to the MDRAW command:

```
1000 xsize% = SCR_XLIM
1010 ysize% = SCR_YLIM
1020 MDRAW #0, menuname, -1,-1, xsize%, ysize%
```


Obviously, you need to allow for any border and shadow, which reduces the available screen area for the menu itself (remember that the size is actually the size inside the border).

This sort of technique could also be used where you want to offer the user a limited range of menu sizes, e.g. a game where the board size depends on the display size selected, but only a limited number of board sizes are available, e.g. 4x4 squares, 8x8 squares and so on. In this case, you'd check the screen sizes if any of the possible selections are greater than the minimum QL screen size of 512x256, then cycle through the selection of sizes when the user clicks on a "resize" icon. If three menu sizes are available, all of which fit in the minimum possible screen sizes:

```
100 board_size = 1 : REMark sizes 1,2 and 3 available

2000 DEFine PROCedure Select_New_Board_Size
2010 board_size = board_size + 1
2020 IF board_size > 3 THEN board_size = 1
2030 SElect ON board_size
2040 =1 : xsize = 128 : ysize = 64
2050 =2 : xsize = 192 : ysize = 96
2060 =3 : xsize = 256 : ysize = 128
2070 END SElect
2080 MDraw #channel%, "mymenu", -1, -1, xsize, ysize
2090 END DEFine Select_New_Board_Size
```

Most programs will need "dynamic resizing" where you click on a "resize" icon, a symbol appears and depending on where you drag and drop this symbol a new menu size is calculated and redrawn as close to that size as possible. This is a bit more complex and is best illustrated with a little program to drive the menu we described above. If we assume that the loose items have the following functions:

LI1: Move (moves the menu around the screen without changing its size)

LI2: Resize

LI3: Sleep in button frame

LI4: ESC (exit from the program)

LI5: Any other function!

Easyptr menus will automatically use a system move if you set the action code for the loose item as the number 5. Alternatively, set a different selection key and use the WMOV #channel% command to call the move routine. If no new position is set in this command, if the parameter is set as -1, it calls the window move routine (window manager move options - drag by outline or drag by window if on a system with Window Manager 2, depending on how SMSQ/E is configured). Thus the command

WMOV #channel%, -1

allows use of the Window Manager window move routine. Whether you set action code 5 for automatic move or use WMOV depends on what your program does. If clicking on the Move icon only needs a simple move, set action code 5 for the loose item and it's all done for you, but if your program needs to take other action before or after the move (e.g. MW_LINKing of windows) you can use WMOV as well.

Dynamic resizing requires the following steps:

1. Get the menu size. The best way is to use the PVAL command after an MCALL to return the relevant values.
2. Use the WSIZE command to call the resizing routine and return the size change as its 2nd and 3rd parameters
3. Add the width change to the existing width to set the new menu size. Subtract the height change from the existing height.
4. MCLEAR to remove the old size of menu from the screen.
5. MDRAW the menu with the new size parameters.

Let's turn this into a working routine.

```
100 REMark demonstration of resizing - needs
    scaling_men in ram1_
110 DIM result%(16) : REMark results from PVAL
    below
120 REMark open a window channel for this menu
130 channel = FOPEN("con_")
140 :
150 REMark draw in original size first
160 MDraw #ch, 'ram1_scaling_men'
170 :
180 REpeat program
190   num = MCALL(#channel)
200   SElect ON num
210     --1 : REMark move
220       WMOV #ch
230     --2 : REMark resize
240       PVAL #ch%, result% : REMark get
        result of pointer request
250       WSIZE #channel, x%, y% : REMark
        resize, size changed returned in
        x%, y%
260       xsize% = result%(8) + x% :
        REMark new width
270       ysize% = result%(9) - y% :
        REMark new height
280       MCLEAR #channel : REMark remove
        old size version
290       MDraw #channel,
        'ram1_scaling_men', -1, -1,
        xsize%, ysize% : REMark redraw
300     --3 : REMark button sleep-automatic if
        action code = 7
310     --4 : EXIT program : REMark ESC
320     --5 : REMark OK-anything you like!
330   END SElect
340 END REpeat program
350 MCLEAR #ch
360 CLOSE #ch
370 CLAMP
```

The part we are interested in is lines 230 to 290. Line 240 calls PVAL to get the result of the pointer request. It returns an array (which should be pre-dimensioned, see line 110) of 16 or more values, the ones we are interested in are elements 8 and 9 which return the window width and height respectively before resizing. Line 250 calls the WSIZE command to do the resizing by moving a resize icon around the screen under pointer control. WSIZE is a new extension in Easypr 4. It takes a channel number parameter and two integer variable parameters into which the size change is returned. Lines 260 and 270 calculate the new menu sizes by adding/subtracting them to/from the size returned by PVAL before WSIZE was called. Note how it's "+x%" in line 260 and "-y%" in line 270. Line 280 gets rid of the old size menu. Line 290 draws the new size of the menu. Note that when resizing, it is possible to change height, width or both. Your program and menu

design should be able to cope with the resultant sizes, be they tall and thin or short and wide! This method gives much more flexibility and choice of sizes than providing a selection of pre-defined sizes in the old way with Easymenu.

I hope that this little tutorial on resizing in the new version of Easypr has been of help. It sets out to chronicle how I went from someone who had no experience of using menu scaling before to becoming fairly confident in its use. Now excuse me, I have a new toy and need to go away and redesign some of my older programs to use this new and much improved facility!

Please note: the facilities described above, such as scaling flags and in particular the new variant of the WMOV command (WMOV -1) are only available in new releases of Easypr 4. They are not usable in Easypr 3. Per Witte has written routines to allow use of scale flags with menus generated by Easypr 3 and these are available from his usual Knoware website.

Keyboards - the missing bit

George Gwilt

Here is the missing listing from last issue's article. The last line was missing.

Table F Setting Program

```

1000 REMark - Set private keyboard tables
1010 :
1020 fch = FOP_OVER (ram1_kbd1)
1022 :
1032 WPUT #fch, 4, 642
1042 :
1210 BPUT #fch, 0,$1B,'1','2','3','4','5','6','7','8','9','0','-', '=', $C2,$09
1220 BPUT #fch, 'q','w','e','r','t','y','u','i','o','p','[','_','$0A, 0,'a','s'
1230 BPUT #fch, 'd','f','g','h','j','k','l',';','$27,$9F, 0,'#','z','x','c','v'
1240 BPUT #fch, 'b','n','m','.',',','/','0','*','0','$E0,$E8,$EC,$FO,$F4,$F8
1250 BPUT #fch, $EA,$EE,$F2,$F6,$FA, 0,$F9,'7','8','9','-', '4','5','6','+', '1'
1260 BPUT #fch, '2','3','0','.', '3','/','\','0, 0, 0, 0, 0,$0A, 0, 0, 0
1270 BPUT #fch, 0, 0, 0, 0, 0,$E9,$ED,$C1,$D0,$D5, 0,$C0, 0,$C8, 0,$C9
1280 BPUT #fch, $D8,$DD,$FC,$CA, 0,$D3, 0, 0, 0, 0, 0, 0,$0A, 0, 0, 0
1290 :
1300 BPUT #fch, 0,$7F,'!', '!', '!', '$', '%', '^', '&', '*', '(, ')', '_ ', '+', $C6,$FD
1310 BPUT #fch, 'Q','W','E','R','T','Y','U','I','O','P','{','}', $FE, 0,'A','S'
1320 BPUT #fch, 'D','F','G','H','J','K','L',':','@','$B5, 0,'~','Z','X','C','V'
1330 BPUT #fch, 'B','N','M','<','>','?','0','*', 0,$FC,$E4,$EA,$EE,$F2,$F6,$FA
1340 BPUT #fch, $EA,$EE,$F2,$F6,$FA, 0,$FB,'7','8','9','-', '4','5','6','+', '1'
1350 BPUT #fch, '2','3','0','.', $A3, 0,'|','$DF,$E1, 0, 0, 0, 0, 0, 0, 0
1360 BPUT #fch, 0, 0, 0, 0, 0,$EB,$EF,$C5,$D4,$D5, 0,$C4, 0,$CC, 0,$CD
1370 BPUT #fch, $DC,$DD,$FC,$CE, 0,$D7, 0, 0, 0, 0, 0, 0,$FE, 0, 0, 0
1380 :
1390 BPUT #fch, $FF,$80,$91,$92,$93,$94,$95,$96,$97,$98,$99,$90,$8D,$9D,$C2,$09
1400 BPUT #fch, $11,$17,$05,$12,$14,$19,$15,$09,$0F,$10,$BB,'@', $0A, 0,$01,$13

```

```

1410 BPUT #fch, $04,$06,$07,$08,$0A,$0B,$0C,$09,$07, 0, 0, 0,$1A,$18,$03,$16
1420 BPUT #fch, $02,$0E,$0D,$8C,$8E,$8F, 0,'*', 0,$FF,$E2,$E9,$ED,$F1,$F5,$F9
1430 BPUT #fch, $D1,$D3,$D7,$D9,$DB, 0,$F9,'7','8','9','-', '4','5','6','+', '1'
1440 BPUT #fch, '2','3','0','.', 3, 0,'\'', $DF,$E1, 0, 0, 0, 0, 0, 0, 0, 0
1450 BPUT #fch, 0, 0, 0, 0, 0,$E9,$ED,$C3,$D2,$D7, 0,$C2, 0,$CA, 0,$CB
1460 BPUT #fch, $DA,$DF,$FC,$CA, 0,$D3, 0, 0, 0, 0, 0, 0,$0A, 0, 0, 0
1470 :
1480 BPUT #fch, $FF,$1F,$81,$A0,$83,$84,$85,$BE,$86,$8A,$88,$89,$BF,$8B,$C6,$FD
1490 BPUT #fch, $B1,$B7,$A5,$B2,$B4,$B9,$B5,$A9,$AF,$B0,$1B,$1D,$FE, 0,$A1,$B3
1500 BPUT #fch, $A4,$A6,$A7,$A8,$AA,$AB,$AC,$9A,$82,$1B, 0,$7C,$BA,$B8,$A3,$B6
1510 BPUT #fch, $A2,$AE,$AD,$9C,$9E,$9F, 0,'*', 0,' ', $E6,$EB,$EF,$F3,$F7,$FB
1520 BPUT #fch, $D1,$D3,$D7,$D9,$DB, 0,$FB,'7','8','9','-', '4','5','6','+', '1'
1530 BPUT #fch, '2','3','0','.', $A3, 0,'\'', $DF,$E1, 0, 0, 0, 0, 0, 0, 0, 0
1540 BPUT #fch, 0, 0, 0, 0, 0,$EB,$EF,$C7,$D6,$D7, 0,$C6, 0,$CE, 0,$CF
1550 BPUT #fch, $DE,$DF,$FC,$CE, 0,$D7, 0, 0, 0, 0, 0, 0,$FE, 0, 0, 0
1560 :
1570 REMark      0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F
1620 :
1630 BPUT #fch, 0, 0, 0, 0, 0,$B5, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
1640 BPUT #fch, 0, 0,$B5, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
1650 BPUT #fch, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
1660 BPUT #fch, 0, 0, 0, 0, 0, 0, 0,$2A, 0, 0, 0, 0, 0, 0, 0, 0
1670 BPUT #fch, 0, 0, 0, 0, 0, 0, 0,$C1,$D0,$D5,$2D,$C0, 0,$C8,$2B,$C9
1680 BPUT #fch, $D8,$DD,$FC,$CA, 0,$2F, 0, 0, 0, 0, 0, 0,$0A, 0, 0, 0
1690 BPUT #fch, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
1700 BPUT #fch, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
1710 :
1720 FOR x=1 TO 64: LPUT#fch,0
1730 CLOSE#fch

```

Internet on the QL – Part I (Emulators)

By Phoebus R. Dodos
B.Sc MIS

1. Introduction

The QL and its derivatives, for quite sometime, have had a host of facilities available to them that are under-used or sufficiently complicated, as to completely discourage regular users from using them.

One of these facilities is the ability to access the Internet. Although Internet access on the QL is by no means complete (and it won't be until soQL or QlwIP are completed), recent advances have made it usable to a degree that warrants, some form of an introduction.

This series attempts to do just that.

We will look into three major areas:

1. Internet (online) browsing
2. FTP access and
3. Email

Although there are more Internet applications than these listed above, these are the most widely used and in one form or another, are available right now to QLers.

This series will be divided in three parts (To start with). This first part will deal with Internet facilities available to QL emulator users and basic setup of Internet applications. The second and third parts, as they are more complicated, will deal with native hardware (Expanded QLs and Qx0 machines) setups.

Topics that will be covered in this series include: native internet access, emulation access (via PC Conqueror no less!), Qx0 Shoestring Linux access as well as, thorough soQL setup procedures among others. I hope that QlwIP will be available for testing by the time this series is almost over, as that is the most complete suite available currently and the best solution for native hardware for broadband access (plus it increases the number of available browsers by one).

Additionally and for the non DIYers out there, all programs discussed here (apart from commercial ones of course), will be available on the web (or via snail mail from me) for free, in both zipped

format and in a qxlwin file that would enable a complete plug and play (at least where application setup is concerned because configuration is another story...) experience.

2. Emulators covered

For most non QL-native (i.e. soQL or QlwIP) access schemes (except the aforementioned solution via PC Conqueror), you will need an emulator that supports some sort of interfacing between the native OS' TCP stack and the emulated operating system. These currently are:

- Q-emuLator (Windows versions 2.3.2 and later in **Expanded** registration mode)
- QPC II v.3.3 and later
- uQLx (all *NIX versions including uQLx-mac but NOT uQLx-win32 presently)

In general for this first part, the emulator needs to have a TCP_ device available to S*basic.

All other emulators (QPC1, Q-emuLator Basic/MAC, QLAY and derivatives and even QLEM or QDOS Classic powered UAE), are considered native hardware for the purpose of this series and will be covered in parts II and III.

Note that, for the aforementioned emulators (the non TCP-capable ones) serial support is mandatory if internet access is desired.

3. Operating Systems/Extensions/Support programs needed

Only one of the applications needs the Extended Environment; namely QPOP3 (For graphical email retrieval)

Only one extension is really needed and that is the env_bin (Environment Variables extension). It is assumed that you will have that LRESPR'd in your boot file.

Although not necessary, Toolkit II (or Turbo Toolkit v.3.x and higher) is recommended as it is useful for the PROG_USE and DATA_USE variables.

Additionally, as it may be useful to ql-lynx users, an editor (we discuss QED here but any editor with ability to take command line parameters - namely the filename to be edited - will do), an image viewer (photon is assumed but also ungif etc is just fine as is the case for the editor) as well as, some unix commands (apart from the ones provided by lynx) are recommended as they will make your life a lot easier.

Operating Systems-wise all programs have been tested with Minerva 1.98, QDOS Classic, JS/JM and SMSQ/e v.2.98 and above. (Q-emuLator has been tested with all OSes including SMSQ/e v.3.09 for the Gold Card).

So if you are ready, let's get down to business with...

4. Web browsing (ql-lynx)

4.a Overview

ql-lynx is one of these programs, that have never been really accepted from QLers although it is working amazingly well and it is extremely stable on the QL. The version that we use is v.2.8.2/jh, available from www.daria.co.uk

One of the reasons for ql-lynx's unpopularity, is the sheer amount of files that will appear once the qlynx.zip archive is unpacked (and that is required by the GPL license which dictates that binaries should be made available together with the source code, although you can in fact break a distribution apart in source files and binaries only - which isn't the case in ql-lynx). In reality however, the files needed are just a few and ql-lynx will work immediately after the correct environmental variables are set.

Moreover, ql-lynx allows access to your email (more about that further below) as well as offline (local) and online (remote) file browsing.

4.b Files and directories needed

The main trick, so you won't be overwhelmed from all the files, is to do a selective unpacking of files; preferably using an archive manager such as ACP

Lynx needs only 3 directories:

1. The root directory (we will name that `win1_lx_`)
2. A help directory (we will name that `win1_lx_h_`)
3. A terminfo directory (we will name that `win1_lx_terminfo_`)

The files absolutely needed for ql-lynx to work, are the following (unpack them all in `win1_lx_` unless otherwise noted)

1. lynx (executable) (Found in the root directory of the archive)
2. lynxrc (text file) (Found in the qdos directory of the archive)
3. lynx_cfg (text file) (Found in two places in the archive - choose the one inside the qdos directory)

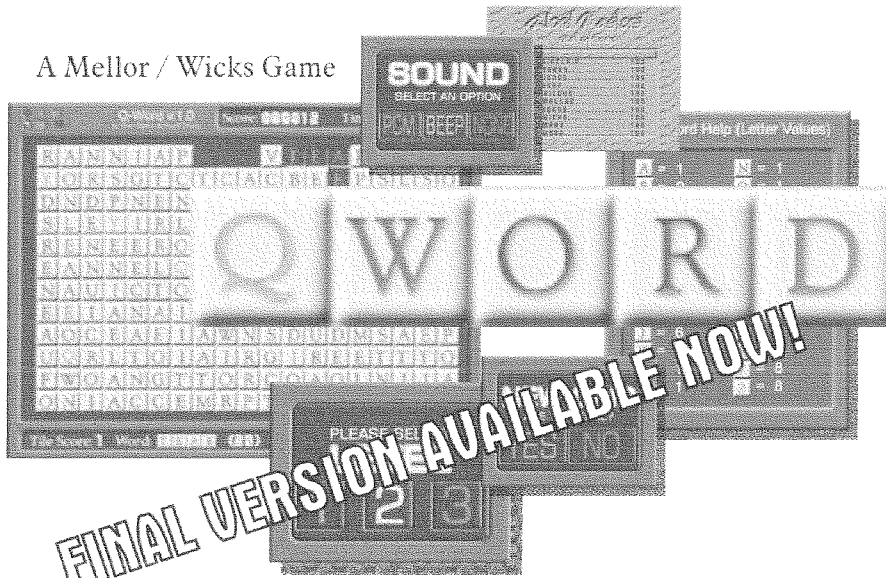
RWAP Software

in cooperation with



PRESENTS...

A Mellor / Wicks Game



for the Q40 & Q60
(And Aurora/SGC)



for QPCII & QXL

FINAL VERSION AVAILABLE NOW!

Scrabble® Meets Tetris® Meets the clock! In glorious colour and digital sound! (Aurora too!)

For the first time in the history of the QL, a truly high colour, digital sound *really* addictive game, in superb 1024 x 512 (640 x 480 on QPC/QXL and 512 x 480 on the Aurora) high colour. SMSQ/e is NOT needed for the Aurora (and soon for the Q40/60) but it is recommended. The innovations are too many to mention but let's just say that the surprises start with its user-friendly installer which is unlike any other!

The object of the game is to clear letter tiles from the grid by using them to form words. Points are based on the length of the word and the letters used. Simple, yet addictive!

PRICES

QWord for QPC2/QXL (CDROM)	£20
QWord for Q40/Q60 (HD FDD)	£20
QWord for Aurora/SGC (HD FDD)	£20
Add Pword (500,000 word UK dictionary)	£10

If you want more than one version of QWord (eg. to play on Aurora as well) - add £5 for each additional version.

SPECIAL OFFER! FOR AURORA/SGC USERS
BUY Q-WORD WITH Colour Drivers version of SMSQ/e for the SGC
!!ONLY £50!!

LANGUAGES

Each version is supplied with one dictionary (normally the OSPD UK English Scrabble Dictionary). However, QWord can also be configured to use the following alternative language grids:

Dutch, English (USA), English (UK), French & German

Specialist QTYP dictionaries for each language have been produced, which removes the need for accented characters in the foreign languages.

You can choose ANY one dictionary to be supplied with QWord.

Alternatively, order additional dictionaries for £2 per dictionary.

RWAP Services

26 Oak Road, Sheffield, Walsall, West Midlands WS4 1RQ, United Kingdom Website: <http://www.rwapsoftware.co.uk>

We accept:



(PayPal/Nochex Payments to RWAPSoftware@aol.com. Cheques in £ sterling made payable to R. Mellor)

4. fork, cp, mv (executables) (Found in the qdos directory in the archive)
5. lynx_hlp (help file) (found in the root directory in the archive)
6. All files under the help directory in the archive. Unpack in `win1_lx_h`
7. qdos (terminfo definition file) (Found in the qdos_terminfo directory in the archive - Unpack in `win1_lx_terminfo`

If you want editing facilities and picture viewing, put qed and photon (or your choice of editor and

image viewer; QD works great with lynx by the way!) in the `win1_lx` directory.

4.c Creating the environmental variables for lynx

The next step to get lynx to work is to create a boot file that will contain its required environmental variables.

The following does exactly that (with a few changes to enable lynx to run in a variety of platforms)

```

100 REMark **Lynx environment variables**
110 :
120 SETENV "TERM=qdos": REMark This changes the terminal type to a VT-52 compatible qdos
      terminal
130 SETENV "TERMINFO=win1_lx_terminfo": REMark This lets lynx know where the terminal
      Definition file is located
140 IF VER$ <> 'HBA' THEN SETENV "LYNX_CON=512x256a0x0_6_7_0": REMark For Q-emuLator and
      Small screen uQLx
150 IF VER$ = 'HBA' THEN SETENV "LYNX_CON=" & SCR_XLIM &"x" & SCR_YLIM & "a0x0_6_7_0":
      REMark for SMSQ/e this sets up a lynx screen at the maximum possible resolution
      looks nice that way) :-)
160 SETENV "LYNX_FONT=win1_lx_pcql_font" : REMark This uses a different font that can
      Correctly render the ISO-8859-1 charmap
170 SETENV "LYNX_CFG=win1_lx_lynx_cfg" : REMark This sets the location of the lynx main
      Configuration file
180 SETENV "LYNXRC=win1_lx_lynxrc" : REMark This sets the location of the user prefs
      file

190 PUSHD
200 SETLYNXD
210 PRINT "TYPE POPD:enter, to restore the data/prog defaults when done with lynx"
220 EX lynx
230 DEFine PROCedure PUSHD
240 defprog$ = PROGD$
250 defdata$ = DATAD$
260 END DEFine PUSHD
270 DEFine PROCedure SETLYNXD
280  PROG_USE "win1_lx_"
290  DATA_USE "win1_lx_"
300 END DEFine SETLYNXD
310 DEFine PROCedure POPD
320  PROG_USE defprog$
330  DATA_USE defdata$
340 END DEFine

```

Note here, that if you are running on a large screen under uQLx (Minerva ROM is required), the "LYNX_CON" variable can be updated accordingly to take advantage of the higher res.

4.d Configuring lynx

QIlynx has two main configuration files: `lynx_cfg` and `lynxrc`. These store the general (global) preferences for lynx and the user preferences respectively. Settings found in `lynxrc`, will override settings in `lynx_cfg` so make sure that you don't change something in `lynxrc` if the default setting in `lynx_cfg` suits you.

Although lynx will run quite happily with the defaults unaltered (as they come from the archive), it is best that we "touch" them a bit, so that we can make the program really usable.

First, (and assuming that you are a complete novice with lynx) you should load `lynxrc` in your favourite text editor.

Find the line (usually line 9) that reads:

```
user_mode=ADVANCED
```

and change it into:

```
user_mode=NOVICE
```


This, will effectively make more options visible while you run lynx and it will make your life easier until you familiarise yourself enough to know, the shortcuts by heart.

Note, that line 14 (file_editor=...) contains already the editor (qed) for use within lynx

The 'fork' prefix is the application that allows qdos executables to accept parameters like in Unix (What it does is to create jobs that are owned by lynx). If you need to change your editor to, say QD, the line should look something like this:

```
file_editor=fork qd_english
```

(Note, that on some QDOS systems, lynx cannot execute anything, unless it is in its progds\$ directory, so copy any applications you need to use with lynx inside the win1_lx_ directory)

Next, save the file and load lynx_cfg into your text editor.

First find the line (usually line 25) that contains the option:

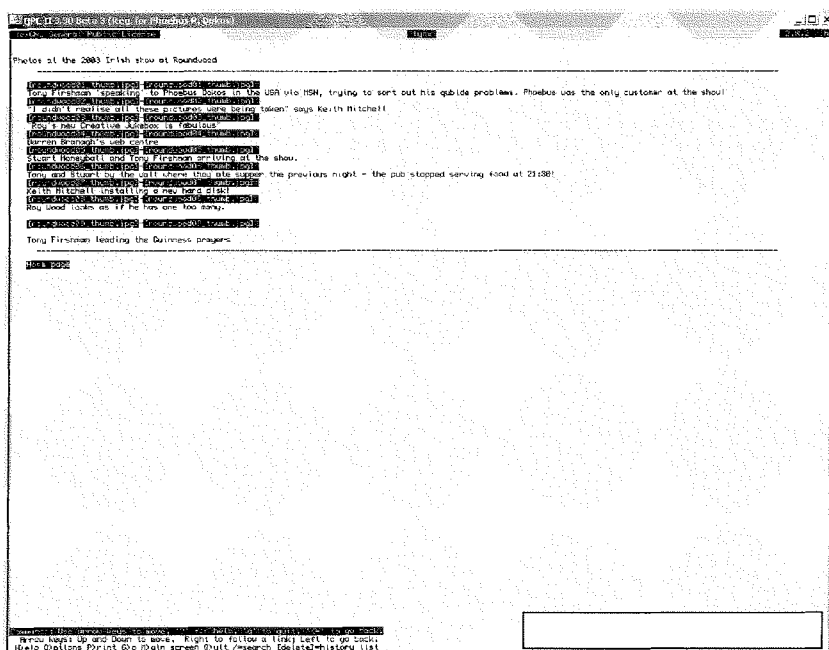
```
STARTFILE:
```

and amend it to use the homepage of your choice. For example, if you want the TF Services website to be your homepage upon lynx's start-up, you should change the line to:

```
STARTFILE:http://www.firshman.co.uk/
```

Next, find the line (usually line 39) that starts with

```
HELPPFILE:
```



Note, that there are a total of 3 lines that have HELPPFILE, but two of them are prefixed by a hash (#) symbol. That symbol means, that the line is commented out (Just in case you haven't figured that out by now). The HELPPFILE option should be changed to point to the correct location of the help files.

It is useful to note, that lynx uses a Unix syntax for its filenames. You should start the line with file:// (to indicate that this is a file and not a url), continue with localhost/ (to indicate that the file is on your machine) and then write the complete path separated by / instead of the standard QDOS/SMSQe separator. Finally if the file has an extension, this should be written as ".extension" instead of "_extension". So in our case, the HELPPFILE line becomes:

```
HELPPFILE:file://localhost/win1/lx/h/  
lynx_help_main.html
```

Once this is done, lynx is ready to run, but still misses the viewers for the various mime file types. To enable these, you should find the lines that start with

```
#VIEWER
```

And change them appropriately. For example, to have photon start, whenever a jpeg file is selected uncomment the:

```
#VIEWER:image/jpeg:xv %&:XWINDOWS  
line and change it to:  
VIEWER:image/jpeg:fork photon %s
```

That will effectively use photon to see the picture, if one is selected (see Figures 1 and 2)

Additional viewers one can use, are sox for sound files (QPC/uQLX only), ungif etc.

4.e Running lynx

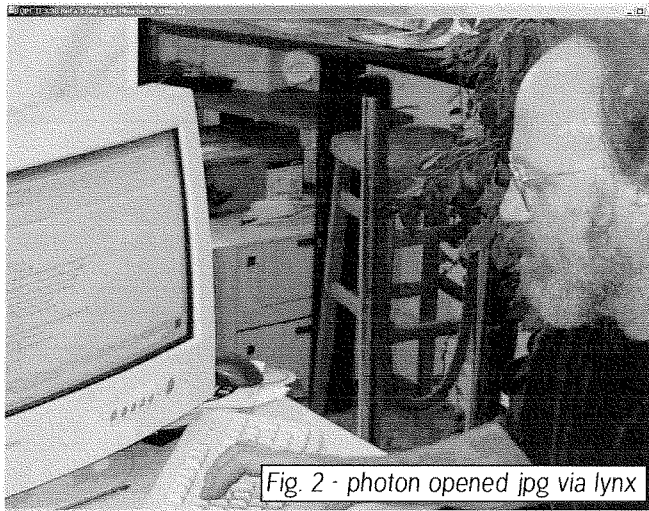
Once all the above are done, you run lynx by either running the boot file we created in 4.c. or by typing EX lynx (Fig. 3 (QPCII) and 4 (Q-emuLator))

Lynx has several command options (which are covered in its manual) of which the most useful, is the ability to go to a URL immediately after startup. For example to go to www.bbc.co.uk, one should type:

```
EX lynx; "http://www.bbc.co.uk/"
```

IMPORTANT NOTE

ql-lynx currently DOES NOT SUPPORT the https:// protocol and therefore secure connections with the QL are impossible. What that means is that you cannot use your QL to buy anything online and it definitely means that all data that you submit to the internet can possibly be seen by others! (At least at present). I hope that by the next article in the series this "problem" will have been resolved



general ftp conventions.

5.c Configuring ql-ftp

Ql-ftp can be configured with the use of a file and an environmental variable, however, unlike lynx, these are not necessary for it to run. For that reason, we will not get deeper into the subject, apart from saying that configuration is fully documented within ftp.txt and readme_qdos within the archive

5.d Running ql-ftp

The simplest command of all is needed to run ql-ftp:

EX ftp

That immediately runs ql-ftp and shows you the ftp> prompt. You connect to an ftp site by issuing an open command such as this:

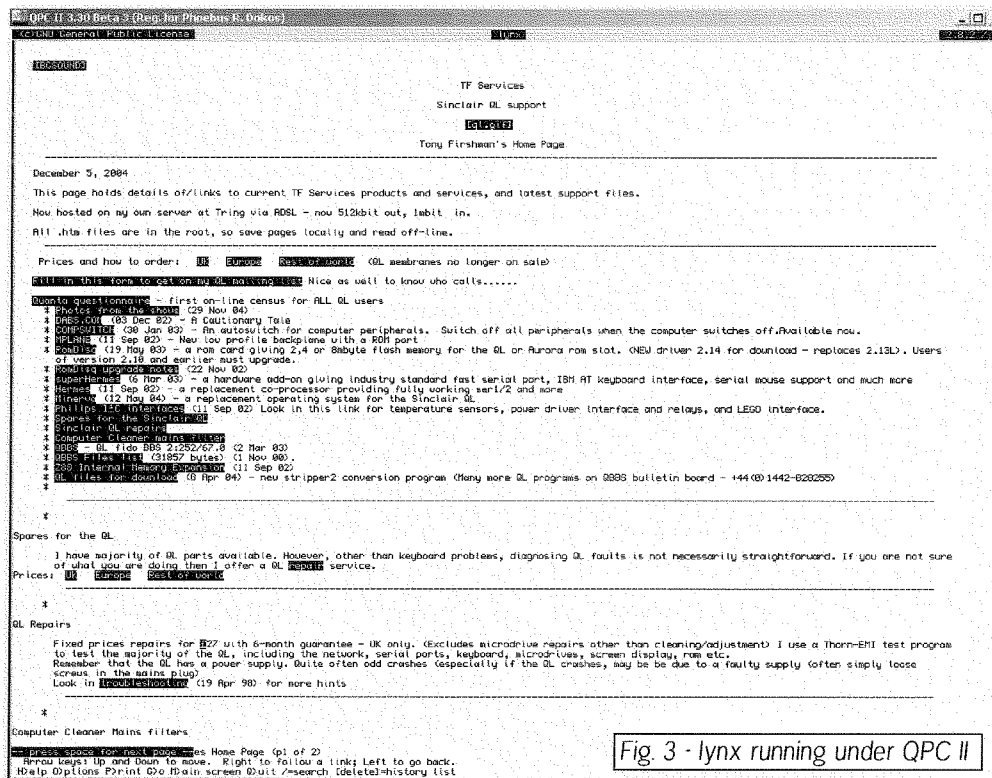


Fig 3 - lynx running under QPC II

5. FTP (ql-ftp)

5.a Overview

ql-ftp is a very easy to use ftp client. For most purposes it is not at all different from Microsoft's or the standard Unix command line ftp clients. Ql-ftp is obtainable by www.daria.co.uk (A good reason to try your brand-new ql-lynx setup too!)

5.b Files and directories needed

Ql-ftp only needs one file: ftp (an executable found in the qlftp.zip archive) and although it is not needed, I would recommend also, to unpack the ftp.txt file from the archive, as it contains a lot of information on its usage and

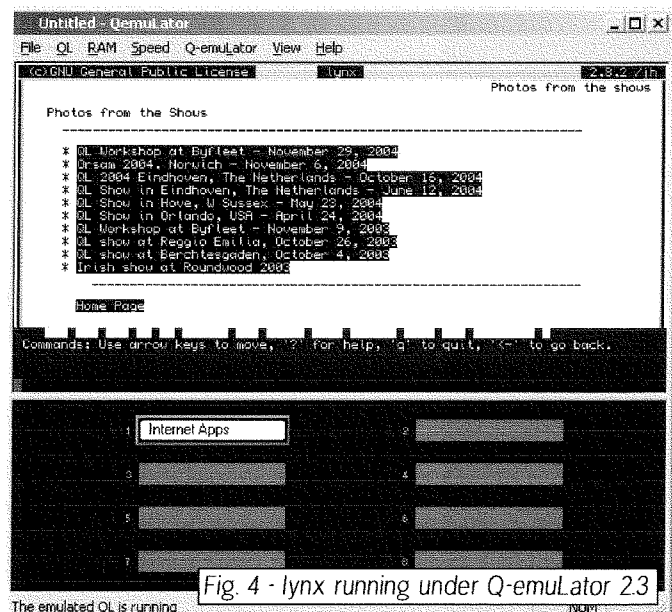
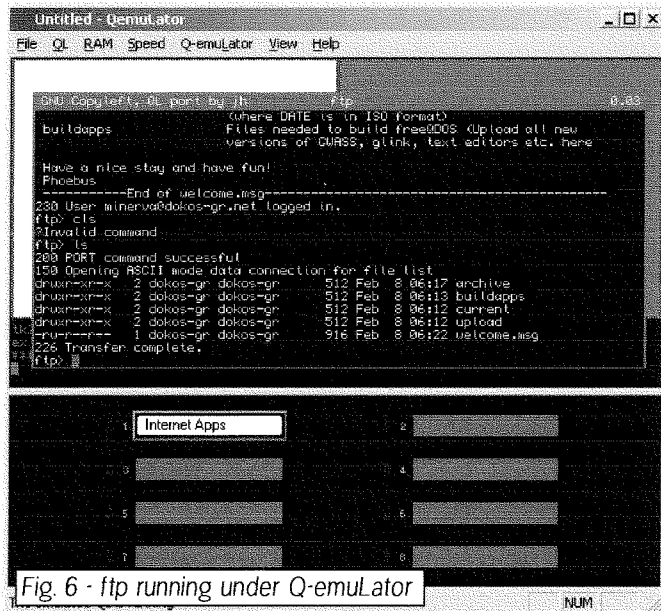
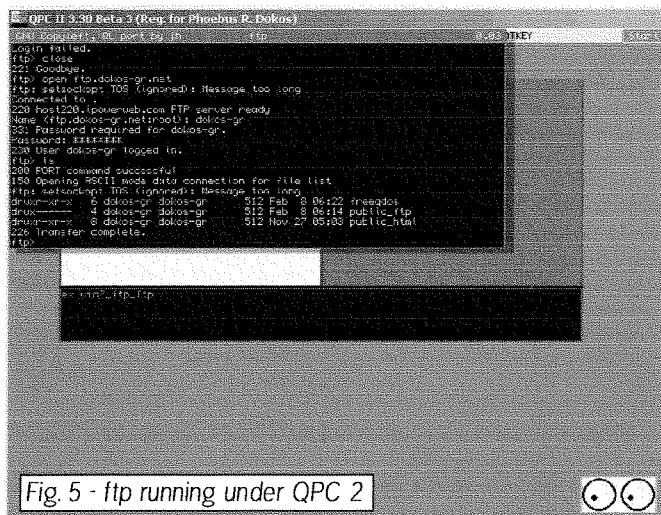


Fig 4 - lynx running under Q-emulator 23

open ftp.dokos-gr.net

Figures 5 and 6 show ql-ftp running under QPC II 3.30 and Q-emuLator 2.3.3 respectively



6. Emailing (fetchpop 1.9 / qmlailer and QPOP3)

6.a Overview

Emailing, unlike ftp and browsing, is a tad more difficult to setup and it's also rather difficult to use (especially sending). However, emails can be read and sent with a bit of effort and as Jon Dent's recent email to the ql-users list demonstrated, this can be even done from S*basic.

We are going to examine the major 3 programs for email, two of which are for reading and one for sending.

The first one: QPOP3 runs well, but may experience issues under either QPC or Q-emuLator if you are using an email virus scanner such as AVG for Windows, so you may want to by-

pass all references to it. The other two (fetchpop and qmlailer) work fine with all emulators so ... read on!

The programs can be found again at www.daria.co.uk in two archives: [qpop3.zip](#) and [qdosemail.zip](#)

Note that fetchpop and qmlailer (the receiving and sending clients) are almost identical to the ones used by soQL so everything you read here will be of use in the next part of the series.

6.b Files and directories needed

We will be putting all three applications under a directory called `win1_email`

For the qpop3 application we will need the following files:

1. `qless` (email viewer executable)
2. `qpop3` (email pop application executable)
3. `qpop3.rc`

For fetchpop 1.9 and qmlailer we will only need two files:

1. `fetchpop` (executable found under the `fetchpop1_9` directory of the `qdosemail.zip` archive)
2. `qmlailer` (executable found under the `mail` directory of the `qdosemail.zip` archive)

As we mentioned earlier QPOP3 needs the extended environment so if you have QDOS or Minerva you should `LRESPRwman` and `ptr_gen` before running the application.

6.c Configuring the applications

QPOP3 needs one file (`qpop3.rc`) and an environmental variable setup in order to run.

The environmental variable, unlike lynx, is case-sensitive and the program WILL NOT run, if either the case or the name are incorrect.

You should create, either a boot file or type in your command line:

```
SETENV "QPOP3RC=win1_email_qpop3.rc"
```

Next you should edit the `qpop3.rc` file to reflect your email settings

The format is

```
POLL[SKIP] server_name user user_name pass  
password
```

Where `SKIP` instructs the program to skip that entry when looking for new mail

Fetchpop is a tad more difficult to configure, especially because it requires a file that starts with a "." By default. Although that will cause you no problems under uQLx; QPC and Q-emulator will protest (internally) so a little workaround is needed.

First you will need two environmental variables named POPRC and MAIL.

The first, instructs fetchpop as to the location of its configuration file which is created automatically upon fetchpop's first execution. MAIL on the other hand, instructs fetchpop the location and name of where to spool the mail file. By definition this is done in ram1_.

You should give:

```
SETENV "POPRC=win1_email_"
SETENV "MAIL=win1_email_mailbox"
```

These, will let fetchpop put its configuration file in the same directory as the application and use the file named "mailbox" as the mail spool file.

Finally in order to configure qmailer, you will need to setup one environmental variable named MAILRC as follows:

```
SETENV "MAILRC=win1_email_mail_rc"
```

and create the configuration file mail_rc. The format for the latter is as follows:

```
name : The real name of the user.
host : your host name (as you'd like in
       the From: line in the email)
smtp : The SMTP host (that relays your mail)
user  : The user name (for the From: line)
sigf  : A signature file (optional)
```

For example, for me, the file should be:

```
name : Ffibys (Hi Dilwyn)
host : verizon.net
smtp : outgoing.verizon.net
user  : pdokos
sigf  : win1_email_sigfile
```

Note that there's no place to write a password. That is because the current version of qmailer doesn't support authenticated accounts. That's not much of a problem, unless your mail server is different from the mail server of your ISP. (The conventional wisdom says, that if you are ALREADY authenticated with your ISP there's no reason to be re-authenticated to send email)

although that is NOT always the case. For most purposes and especially for standard modem connections that is indeed the case however.

6.d Running the applications

QPOP3 has very good documentation in the form of an html file (and a very good excuse to try your new lynx setup) so no specific instructions will be provided here other than how it runs:

```
EX win1_email_qpop3
will suffice (provided that the EE is loaded and
the environmental variable set)
```

On the other hand fetchpop, as mentioned above, needs a special first run in order to be usable.

Provided that its variables are set, you should give the following for first execution:

```
EX win1_email_fetchpop; ' -f fetchpop_rc'
```

Doing so will try to find a file named fetchpop_rc (and since it won't, will create it instead) which will be the default file next time you start fetchpop. When fetchpop starts for the first time, it will ask you for the name of your pop3 server, your user name and password and store them in the file you defined previously. (Figure 7)

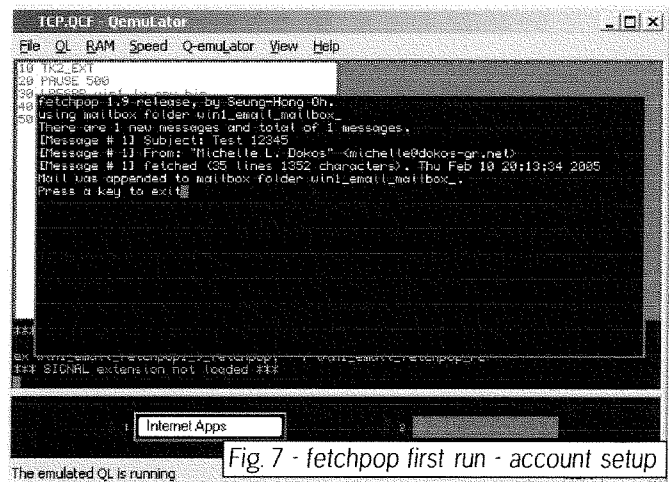


Fig. 7 - fetchpop first run - account setup

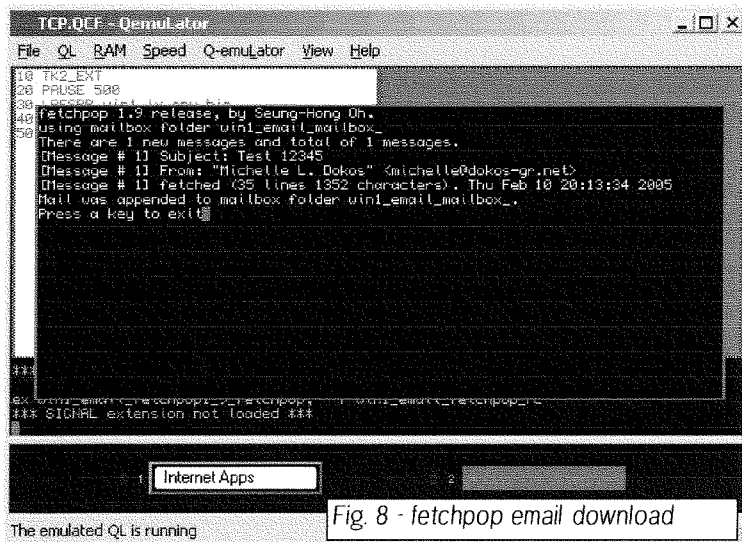
It will then store the new emails into your previously defined mail spool file (if the file doesn't exist, it will create it) (Figure 8).

On uQLx fetchpop can then be run from then on with simply a:

```
EX win1_email_fetchpop
```

On the other emulator systems however you should always use

```
EX win1_email_fetchpop; ' -f fetchpop_rc'
```



'win1_email_mail_rc' is the full path of the MAILRC file
'win1_lx_ql.m' is a log file."

7. Conclusion

As you saw above, with a little trial and perseverance you can have a quite usable Internet suite in your emulator. Moreover (as we will cover later in this series), the TCP capabilities of modern emulators can be used for extremely fast transfers between Emulator-running computers that are on a network. Using sernet-UDP (a nice hack to the original sernet, by Marcel Kilgus) you can

There are more options to fetchpop than the configuration file of course, however the manual (`fetchpop_txt`) does an excellent job of listing all these and you should refer to it to see the complete array of available commands.

Qmailer is simpler to run, but complicated in its options.

First of all, qmailer **does not** have a mail editor. You should therefore, create a text file in advance (for example `mail.txt`) that will contain your message.

Provided that your configuration file is named `mail_rc` (as above) in order to send an email to ql-users (`ql-users@q-v-d.com`) you should do the following

```
ex qlmailer;'-v -f mail_rc
ql-users@q-v-d.com <mail.txt'
```

will send an email written in `mail.txt` to ql-users using the smtp server defined in `mail_rc` (That is to show that you can have more than one smtp servers defined; for example if you have more than one ISPs)

Quoting from the qmailer readme:

"qmailer can be used as the mailer for lynx, so 'mailto' links work.

Simply set the following entry in lynx_cfg:

```
SYSTEM_MAIL: win1_email_qlmailer -vv -f
win1_email_mail_rc 2>win1_lx_ql.m
```

Where 'win1_email_qlmailer' is the full path of the qmailer program

exchange QL files directly between QL emulators locally and remotely. Moreover, with the advent of a more complete soQL and the excellent qlwIP that can happen between all QL platforms available to us today. In the next part we will deal with regular QLs and non TCP-enabled emulators and use a few tricks to enable us to use the internet even from a simple (Trump card minimum) QL

Of course these solutions will work with all emulators as well...

8. Links

The files described here were downloaded from:

1. <http://www.daria.co.uk> (Jonathan Hudson's site: QL-LYNX, QL-FTP,QPOP3, QDOS EMAILER)
2. <http://thgodef.nerim.net/qdos/english/index.html> Photon, QED

Emulator websites

1. <http://www.kilgus.net>: QPCII
2. <http://users.infoconex.com/~daniele/q-emulator.html>: Q-emuLator 2.3.3
3. <http://uqlx.dokos-gr.net> uQLx - mac (Follow the links for uQLx for *NIX)

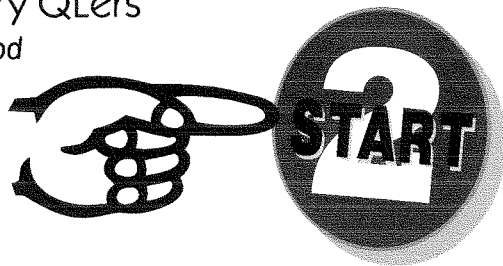
Additionally qxlwin files and compressed archives ready for your emulators can be found at <http://uqlx.dokos-gr.net/ql-net/>

Next issue, I'll deal with real QL hardware and internet access.

Start Here

Essential Information For 21st Century QLers

Roy Wood



PART II Drive Time

Disks Don't Have to be Hard

So now, after the last issue, you should have your screens how you want them. I have deliberately left the issue of colours and colour drivers to a later section of this series because I feel the next item on the agenda should be storage.

It is time to look at the next problem which besets many newcomers to updated hardware. Taking your first steps with a hard drive can make some people break out into a cold sweat and I know of at least one user who has hardly used the drive on his Q40, preferring to BOOT from floppies as he always did.

First a bit of history.

A Miracle!

I think that the first QL hard drive was probably the Miracle systems one. These were selling for 449 pounds in the January 1990 Edition of QL World. They had 40Mb of storage and plugged directly into the ROM port at the back of the QL. Hard drives in those days were either of the MFM or RLL type. They were rather slow and required the user to 'park' the drive heads at the end of each session so that the drive platters were not damaged on startup. This was followed by the Falkenberg Hard Drive Interface. This interface took a different route to connection because it required that you open the case of the QL remove one of the chips and plug it onto the board in its place. That chip is then inserted into the interface board. (Stuart Honeyball of Miracle Systems always maintained that the control code for this was ripped off from his original design). The interface also used the MFM and RLL drives and connected to them by using a standard PC drive controller. This was the interface being sold by Qubbesoft and W.N. Richardson when I took my first steps away from the Black Box and was the one which was in the tower case system, assembled by David Johnson, which was my first expanded system.

This was a reasonably efficient system and served me well with very few problems. One quirk it did have was to rename the floppy device on boot up so your boot file had to have a FLP_USE FLP command at the start if you wanted to use the floppy drive. This system lasted me until the arrival of the Aurora when I discovered the second problem with the interface. The code used to control the drives was about 10k larger than it claimed it was and this 10k was in the address at the top end of the QL screen driver. with a 512 x 256 screen this was no problem but, with the Aurora's extended display, I wound up with a small area at the bottom of the screen corrupted by the interface's code.

The Qubide, designed by Zeljko Nastasic and built and sold by Qubbesoft, had been around for a couple of years and has probably been the most successful Hard Drive Interface for the QL. The software for it was written by Phil Borman using routines converted from the old Rebel Drive. Many people still use this and there is still a demand for second user items. Nasta is currently working on a SuperIde with many new features but I will talk of that later in the article.

Quite aside from the hardware side of QL hard drives there are also the systems in use by the emulators. All of the QL emulators make use of the hard drive on the system on which they are run. Finally we have the Q40/Q60 which has a hard drive system too.

As you can see there are many different ways of having a hard drive on your QL system. There are also nearly as many ways of handling and dealing with those systems. The purpose of this section is to help you to get the best out of some of the more popular ones.

Devices

QDOS and SMSQ/E see the storage media in the form of 'system devices'. That is to say each type of storage medium has a distinct name. There can be eight of each of these per device. The Original black box had only MDV1_ and MDV2_ but, had you used the expansion connector at the side you could have gone up to MDV8_. Toolkits and software extensions introduced the dynamic RAM Disk and you can create up to eight of these two. If there was any way to connect them you could have up to eight floppy drives but the extension board that Miracle made to connect another two floppy drives to its Gold Card was never that stable.

It follows, therefore, that you can have up to eight hard drives as well. (There are drawbacks in doing this for native QL hardware on the Qubide but I will discuss that later). Most physical hard drives

can be divided into partitions. This means that you can take a 10Gb drive and divide it into chunks of whatever size you want. On the PC this is done using software such as Partition Magic and once a partition is created and formatted it appears as part of the drive table and is given a letter.

QL hard drives are all named WINx_ after one of the original hard drive systems, the Winchester Drive. Modern QL hard drive systems do have one useful property however, you can hot swap the 'logical' drives in software. This means you can have several drives available but only link in the ones you want. You can even re-arrange the order of the drives so you can have different BOOT files and swap them in and out. Again, I will come to this later in the article.

Why do we use Hard Drives?

I suppose the answer to that question is speed and convenience. When you consider that an HD floppy drive will hold just under 1.5Mb of data or code, a small (by today's sizes) hard drive will hold 30Gb and the access time for most of these drives is measured in milliseconds you will see that there is no way that a floppy can compete. If you have a double disk drive with the program in one drive and the data in the other the system is going to have to spin up the first drive to load the program and then the second to load the data. If you then want to combine this data with another set of data on another drive you have to remove the first drive and insert a second one and wait for that to spin up too. A 256Mb partition on a Qubide or any other QL emulator, (hardware or software based) can hold all the programs ever written for the QL and probably all your data too. Tony Firshman did tell me that he had completely filled much bigger drives with QL programs and data but, when questioned on this, he did say it was duplicated a few times as backups. Ah yes backing up. That is something else we will have to cover in this section.

It is not only the speed and accessibility you get that should make you lean towards a hard drive system. The other advantage is, of course, that you can set up a 'system' and that is something we will be tackling in this series of articles.

If you are running on single drives you can only load a few programs at any one time. You may have a BOOT file on the master disk which will load, say, SMSQ/E, QPAC 2 and QD and then have a bit of space for one of two smaller programs which can be run later. Suppose you want to run something that is not on that disk. Well you have to take out the first disk and insert the second. If you are lucky the extensions needed for

the second program will already be loaded by the first disk but, if not, you will have to load them either by running another BOOT file and maybe wiping out the first, or by hand from the command line. This, of course, is how we all did it back in the 80's when we had only small drives to play with. Most people would have one program on each disk, or even microdrive, and just change the disk and reboot.

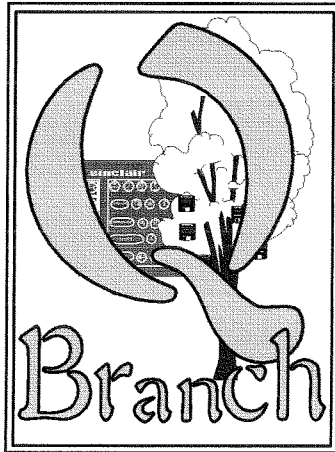
If you have a hard drive you can decide which extensions you need and load them all at the start (provided there is sufficient memory). You can also have every program you need and all the data all there, ready to run. This does lead you into two, completely separate areas. To start with you have to be able to construct the BOOT file, a task some people find slightly more daunting than wading through fire ants in flip flops. You also have to be aware that all of your eggs are in one basket and, if that basket breaks (i.e. the hard drive fails) you have not even got an omelette! One word BACKUP - don't say I didn't say it more than once.

And Now The Hard Stuff

I think we can safely ignore the Miracle and Rebel hard drive systems because it is unlikely that any of you will have them laying around and even less likely they will still be working. I did hear of a couple of Miracle drives still in use a while ago but those who have a working one will know how to use it already and, to be honest, I have not much idea about them myself. As I said before I did have a Falkenberg interface which worked well apart from a few minor niggles but the RLL and MFM drives and interfaces these need would be hard to source so I think we can stick with the three major hard drive systems still in use. These are the Qubide card for QL and Aurora Systems, The Q 40 hard drive system and the QXLWIN files used by the QXL card and QPC2. The Q40/Q60 can also use the Qubide Code to run its hard drives since the program was made Open Source a few years back. Some of the other emulators may use other systems for their drives but, again, this is outside my experience. If users would like to contribute articles on the QEmulator and QLAY systems we would be happy to include them in this series.

Qubide

The Qubide is a small board designed to be plugged into the expansion slot at the end of a QL board. There is another expansion socket on the other end of the board so that the user can plug a Gold or SuperGold Card into it. You can also use a Qplane which will allow the user to



Text 87

£ 79.00

Typset 94 £ 29.00
 Fountext 94 £ 39.00
 2488 drivers £ 29.00
 Epson ESC/P2 drivers
 £ 26.00

Text 87
 High Colour
 Patch
 Available Now
£9.00

PROGRAMMING

QD 2003	£ 49.00
QD + QBasic	£ 63.00
QD + Qliberator + QBasic	£ 104.00
Qliberator	£ 50.00
Master Spy v 3.3	£ 30.00
QPTR	£ 32.00
Easyptr pts 1 & 2 v4	£ 41.50
Easyptr pt 3 (C library)	£ 14.00
QMake	£ 18.00
QMon / JMon	£ 22.00
Basic Linker	£ 22.00
DISA 3	£ 34.00
QMenu	£ 16.00

More New Stuff

This is really turning into a bumper QL year. This month should see the release of a drastic upgrade to EasyPTR. Marcel Kilgus has been hard at work making EASYPTR compatible with the new high colour systems available in SMSQ/E. He has drastically overhauled EASYMENU and given it many new features and he has also updated many of the keywords used by the program. Too much to tell you here!

Not content with that burst of work he has also completed QPCPrint which should be available by the time you read this.

Print to any printer from QPC2
QPCPrint £ 27.50

QDT

£ 30.00

+ P&P

£2.00 (UK) £2.50 (EU) £3.00 (ROR)

Requires SMSQ/E to run

Buy QDT and any copy of SMSQ/E
 and get £10.00 discount

EasyPTR v4

High colour support !
 Major overhaul !

Full Program Pts 1&2

£ 41.50

Upgrade from any Version

£ 27.50

SMSQ/E

Gold Card / Atari / QXL Version

Version 3.10 out now! **£ 32.00**

Aurora High Colour
 version available now

£ 56.00 upgrade **£ 21.00**

Get Cueshell for only £15.00 with any copy of SMSQ/E

UTILITIES

FiFi 2	£ 21.00
QSup	£ 30.00
QSpread 2003	£ 51.00
Cueshell 2	£ 15.00
Qload / Qref	£ 15.00
Disk Mate 5	£ 16.50
QPAC 1	£ 22.00
QPAC 2	£ 42.00
QTYP 2	£ 31.00
QLQ	£ 30.00

QL Keyboard Membranes

Available Now !

£ 17.50 ea + P&P

Q Branch

Feeling out on a limb?
Reach out for Q Branch.
Suppliers of Quality QDOS/MSQ products
Hardware and Software.

Tel +44 (0) 1273-386030 fax +44 (0) 1273-430501

Mobile +44 (0) 7836-745501

email qbranch@qbranch.demon.co.uk web : http://www.qbranch.demon.co.uk

20 LOCKS HILL, PORTSLADE,
E. SUSSEX. BN41 2LB. UK.

Q Branch

Hardware in Stock Now!

Second User Qubides, Gold and Super Gold
Cards: In stock!
Second User Black Box QLs, tested and
working: In stock!
New Aurora Boards : In Stock!
Many more Items. Call for details.

QPC 2 v3.31

Now Only

£ 65.00

New Version !!

QPC Print Out Now

These programs are now
High Colour

QMake

WinEd

QPAC 1

QPAC 2

QSUP

<<<<<<NEW>>>>>>

SuQcess

Disk Mate 5

Please send old master
disks and £1.60 to cover
return postage and copying
fees.

Hardware

Second user items are sometimes available
from us.
Call for details.

R'cycld SuperGold Card	£ 110.00
Recycled Gold Card	£ 45.00*
Recycled Aurora	£ 65.00*
Qubide	£ 45.00*
Qplane	£ 5.00
Recycled superHermes	£ 65.00*
Aurora cables	£ 3.00
Aurora rom adaptor	£ 3.00
'Arfa Braquet'	£ 8.00
MC plate	£ 6.50
New Aurora M/b	£ 70.00
New Aurora M/b	£ 70.00
w 8301 & Minerva	£ 80.00

Qubide upgrades

to version 2.01 £ 8.00

* when available. Call for details

These Programs are paid Upgrades:

QSpread 2001 to 2003	£ 10.50
QD 98 to 2003	£ 10.50
QPC2	-
update from v3.xx	Free
update from v2	£ 13.90
update from v1	£ 34.00

Return Master Disk
With Order

Q Branch Programs

The Knight Safe 3	£ 35.00	Q - Route v1.08C	£ 25.00
upgrades from previous versions	£ 5.00	Route finding programme	
Q - Count	£ 25.00	The Fractal Collection	£ 35.00
Pointer driven home accounting			

Post and Packing is included with Software Prices
Unless Otherwise Stated.



We can accept payment by MasterCard, Visa, Switch, and Maestro. We can also
accept sterling cheques drawn on a UK bank or payment direct to our UK Bank.
Barclays Bank Account# 50543497 sort Code 20-12-75 Acc name QBranch



mount the Qubide and expansion card vertically above the motherboard or, better still, an Mplane. I will discuss these later.

The Qubide has jumpers which will need to be set correctly for it to work. Settings for these are in the Qubide manual and should be studied before the device is installed and powered up. The voltage jumpers are very important. It can support two IDE devices on a single cable plugged into the single IDC header on the board. The cable itself must be a 40 wire IDE cable with all of the wires connected. It will pass straight through the first connector and onto the second. No cable cuts or twists are needed. The devices are selected by means of small jumpers mounted on the rear of the drive. Most drives have three settings, Master(M), Slave(S) and Cable Select (CS). The main drive you are using should be set to master and a master drive has to be present for Qubide to work. You will usually find a diagram on the drive itself which will show the jumper settings. It is also important that, if only one drive is used, that drive should be the last on the cable. This prevents the end of the cable radiating the signal and causing drive errors.

When the QL is fired up it should display the message:

```
Qubide/RebelDrivervx.xx Initialising....  
followed by the drive name.
```

A message saying 'No Drive 0 Found' usually means either the drive is not working or is not connected correctly. It can, however, also mean that the drive is not recognised by the Qubide. All IDE drives have small chips on the circuit board which control the drive. They do much more than just run the physical actions of the drive. They have unique ID strings which identify the drive type, size, number of heads, sectors etc. Apart from that they also map out the bad sectors and mark them as not being available for writing. All drives have bad sectors and are programmed to be tolerant to a low percentage of these. If a drive gets damaged it may start racking up these bad sectors and then the drive becomes increasingly useless until it fails completely. Modern drive controllers are dynamic in the way that they order the data. So much so that it is the drive controller and not the operating system that knows where the data is stored. This gives rise to many interesting problems but this is not the place to discuss them (and I am not sure I have enough knowledge to really give a detailed description).

When the Qubide was first made hard drives of 20mb were expensive and anything larger was barely envisaged. These days 40Gb is regarded in the PC world as being kids level and Terrabyte drives are beginning to appear. This leaves the

Qubide in a dilemma because it is trying to see drives it was never designed to see. It is a tribute to the good software and hardware designer, Phil Borman and Zeljko Nastasic, that it can see any modern drive. I have not really tried many of the more modern drives on the Qubide but it was able to see some of the 20Gb ones I had here.

Once you have the drive recognised by the system the next step is to format it. Qubide will mount the drive as WIN1_ when it is first presented with a drive and it is only WIN1_ that can be formatted from BASIC. The Qubide disk does come with a program called 'PARTITION_EXE' which is the key to creating partitions and formatting them. If you want to have more than one partition on the drive you have to use this program to create them.

Why Do We Need Partitions?

The simple answer is that the Qubide driver was never set up to use more than 256mb. Using a larger size of partition could cause problems. There is another factor to be taken into account. Every active partition on the system will use some of the system RAM to store the File Allocation Table (FAT). The size of the table is determined by the size of the partitions that are active at the time. If you are using all 8 win drives and have them all at the maximum 256mb you are using a sizable chunk of the system RAM on a Gold or Super Gold Card System. It was partly to help avoid this situation that the idea of 'paging in' drives was conceived. You can create many partitions on the drive and then just use the ones you need. I will come to this later.

So the first task should be to create the partitions. Look at the drive size and decide how you are going to use it. When you are creating the partitions you should really name them too as this will help you when you are deciding which ones you are using. Any names you give to the partitions will show in the top right hand window of the QPAC2 files menu so it is easy to read.

You can also have two different block sizes. Block sizes are a determining factor in the drives File Allocation Table. To quote from the Qubide manual at this point:

The driver supplied on the ROM works with two different block size options. A block size of 4 limits the maximum size of a partition to 128 Megabytes, and sets the "file allocation unit" to 2K (4 sectors of 512 bytes each = 2K). A block size of 8 sets the maximum partition size to 256 Megabytes and the file allocation unit to 4K You can use a mixture of 4 or 8 block size on different partitions.

You are given a choice as the block size represents a trade-off of memory usage against partition size and storage efficiency. A small block

size gives better disc usage (less roasted space) but occupies more memory for the map than the same size partition with larger blocks, and a small block size also reduces the maximum partition size available.

As a recommendation, if you have 2 Megabytes of memory or more (Gold Card or Super Gold Card), but your disc is smaller than 128Mb, a block size of 4 will work well for you. If you have less memory than this, 8 will probably be a better option unless your drive is less than 64Mb. If your disc is larger than 128Mb but less than 256 you can use it all as one drive by selecting a block size of 8, or split it into smaller partitions as you wish. Over 256Mb you will have to use partitioning, or only use part of the drive.'

The partitions that are created by the program underwent an overhaul when the firmware on the ROM changed to v1.54. Users who upgrade from that ROM version to any other version have to run a special conversion process to change the way the partition is set up. The program itself is called UPDATEMAP_OBJ. If you are considering an upgrade such as this the first thing to do, before getting the upgraded ROMs, is to make a thorough backup of the entire system. I will come back to backups later in this section.

Instructions on using the Partitioning program can be found on the Qubide disk so I will not go through them here. It is important to use the correct version of the Partition program. From ROM v2.00 this changed and you can find updated programs on the disk provided with the upgrade. It can also be obtained free of charge by sending a disk and return postage to QBranch. New options were added to the program in V2 and can enable their use on removable drives. It is certainly possible to create Qubide partitions on Iomega ZIP drives but they do have to be the internal IDE variety. Those that connect via a parallel port will not work.

The commands which are used to load and unload partitions are loaded by the ROM on initialisation. These commands are in the format:

WIN_DRIVE x,x,x

They are, in fact common to the Q40/Q60, QPC1, QPC2, and QXL – so I will tackle their use and syntax at the end of this section.

Faster Boots - superHermes Maybe?

When the Qubide first starts up it looks to see what is attached to it. This is what is going on when the message 'Initialising....' is displayed. At this point it is polling the drives on the end of the cable and reading the information on the drive controllers. When it has read that information it will print it to the screen set up the drive parameters

and the process of booting the system can continue. This is a bit time consuming if you always have the same drives connected so a way was found to get around this delay.

Minerva MkII has a small section of programmable memory and it was realised that one use for this would be to store the Drive IDs. When the system has booted you can run a program called 'storeI2C_obj' which you will find on the utility disk. This is a small menu driven window which will allow you to write the drive values to the I2C PROM. To use this you will have to have the file 'I2C_IO_REXT' loaded. This is supplied with Minerva MkII. Next time you boot the system Qubide will read these values and not look for any other drives.

A similar thing can be done with the full version of superHermes (superHermes Lite is missing the memory required). The program for this is called 'storeSH_obj', somewhat unsurprisingly. To use this you will need to have the 'IPCEXTUK_BIN' file loaded. This is also supplied with the hardware.

Be aware that, if you change the drive and do not clear this memory, the Qubide will not function. Clearing the memory of both of these processes can be easily done by running the relevant program from a floppy disk and selecting the 'CLEAR' option.

Drawbacks and Failures

Life in the garden is not always as rosy as it could be and there are occasions when the Qubide has problems. That said it is still the most stable of all of the QL native hard drive systems.

The problem many people have come across is the 'FAT is Wrong - Try Rebooting' message on startup. This occurs when the Qubide encounters a corrupted File Allocation Table. Sometimes a simple reboot may clear the problem but sometimes more drastic action is called for. Some of these problems occur because the Gold and Super Gold Cards are inherently electronically 'noisy' and this 'noise' rattles around the IDE bus and corrupts the data. Cleaning the contacts sometimes helps.

It also often happens when the SuperGold Card and Qubide are attached to the mainboard via a QPlane backplane. This is essentially a vertical circuit board which will take the expansion port of the motherboard at the bottom and has slots for the other devices above it. Although this was good in its day it is not as electronically filtered as the Mplane. Switching to an Mplane can often solve the problem.

NEXT ISSUE

QPC2 / QXL / Q40/60 Hard drive systems, Backing It Up, Keywords, and Utility Programs.

Program Development in Assembly Language - Part 14

Norman Dunbar

For the next couple of issues, I'll be going through the way I tend to write my assembly language (and indeed, all my other languages too) programming from the initial thought to the 'final' completed program. I put 'final' and 'completed' in quotes because programs never ever reach that stage there are always bugs to fix and improvements to be made.

Talking of bugs, in the article in Volume 9 Issue 5 (Jan/Feb 2005) I have a couple of fixes to point out in the recursion article.

First of all, on page 48 on line 1050 of the SuperBasic test harness for the factorial routine the range should be 1 to 9 and not 0 to 9 as the error message indicates. Entering a zero would cause a lockup as the stack wandered all over the memory of the system.

Next up, on page 49. Just above the title line for 'The Fibonacci Series' there is a strange bold line centered on the page. It states 'YOU ARE HERE - THE ABOVE IS TESTED OK'. This is an indication to myself that everything in my original article text above that point had been tested and passed. Everything below had yet to be fully tested. I usually remove those indicators when I'm done, but that one must have slipped through. And yes, the code below it was tested before I sent the article to Dilwyn & Jochen!

[... and I was wondering about this when I was laying it out but thought it must be important; another mystery solved - Jochen]

Finally, on page 50, line 1050 of the SuperBasic listing should prompt for a number that is 0 or higher and not just 0 to 9. The joys of reusing code without changing error messages! That is all I have on my previous bugs, so on with the article on program development.

The Initial Thought

The initial idea for a program comes at the most inopportune moments I have found. I've had 'great' ideas at three in the morning, at other times when I was in the bath reading a novel, while driving to work and so on. The fact is, you never know when an idea will suddenly appear, so be prepared and have a bit of paper and a pen handy - not while driving of course - to jot down your ideas before they vanish from memory forever.

Work It Out

Sometimes, given a little thought, the initial idea is found to be not so good after all and the project is abandoned there and then. Those ideas that get through need to be fleshed out a little to see just how good they are.

If they get past this stage, we can start to jot down the basic structure of our program. I personally tend to start with 'the big idea' and break it down into stages before breaking these down into smaller stages and so on until I have a set of small (hopefully) self contained routines at the bottom. This is top down development and used to be quite popular.

Start Writing Code

At this point, armed with your list of routines, you can begin to write down your initial thoughts for the code you want to write to make the 'big idea' come to fruition. Having all the routines broken down by the previous stage, you know where repeated code can be extracted to a sub-routine and so on.

I tend to use a pencil to write code at this stage and arm myself with a decent rubber (eraser for my American readers!) because mistakes will be made. I also arm myself with three books:

Andy Pennell's QDOS manual and his Assembly Language Programming book plus my trusty copy of the Motorola MC68000/MC68008 Programmer's Guide. I also have a cheap narrow feint ruled A4 sized notepad to do my coding on. I then let my brain run away with itself to see how many different mistakes I can make in as short a time as possible.

Even after all these years, I still write down assembly code that just isn't legal syntax and this is sometimes 'obvious' when I look over the code, but usually I notice when George's trusty assembler (GWASL) complains about something in my code.

As I produce code for one routine, I usually find myself needing another, so I note it down on my list and carry on. This 'stepwise refinement' of my rough draft usually produces code that will be typed in using my trusty PFE text editor. This isn't a QL program, it runs on Windows, but I've used it for many years to write code and I prefer it. It allows me to save code in Linux format - which just happens to be the same as the QL's format and I like it.

Once I have the code typed into a file, it gets saved to my C:\ or D:\ drive ready for import into QPC. Within QPC, my code files are copied from the DOS_ device to my RAM_ disc and GWASL is called into action. It almost never assembles first time.

QED is fired up and I make my changes to the RAM_ version, saving the file to DOS_ as a backup. Once I have a code file that actually assembles, I save the whole lot to WIN1_SOURCE_ and get ready to test it all out.

Testing The Code

I tend to look on the bright side of most things, and running my own code is always fun. I simply EX the binary file and see what happens. Usually, it's a crash or system lock-up and I have to reboot. At least rebooting QPC takes a lot less time than rebooting Windows.

So, I know that there is at least one bug in my code and so it's bug hunting time again. After reloading, I run my next test with a code listing and JMON/QMON to trace through the code.

I wrote an article recently about debugging with JMON/QMON

so I won't go into great detail here. Suffice to say, my initial trace starts off with me single stepping up to each sub-routine call, then let each sub-routine run as a single unit. This way, I tend to quickly find out where my major problem lies.

After another reboot - if required - I use the procedured outlined in my JMON/QMON article to set a break point at the 'broken' sub-routine, and I run the code to that point. From there on, I trace the code one line at a time until I hit a sub-sub-routine and let that run as a unit again. Once more, I quickly narrow my search for the main problem down to a single (or a couple) of small bits of code.

This code is then breakpointed and tested again, but in single step mode all the way through. Eventually I either find the offending line(s) and fix them, or I find out which conditional branch I've got the wrong way round - I have been known to BCC when I should have used BCS and so on.

The rest of the process is similar to the above. It may not be the best in the world, but it works for me and I can quickly get debugged code finished and start 'tarting' it all up.

To show how easy most of the

above is, I am going to work through a full example of 'an idea' from initial rough draft onwards to the finished code. I'm still working on this code at the moment and will not be writing up the article until I'm finished. I shall be documenting the process as I go and will write the article up from that.

You will no doubt have read some of my rants and raves about the Disassembler I'm writing as a project for this series. It has been developed bit by bit without any of the above 'discipline' so it has suffered from an extremely large number of errors, some stupidity on my part and a couple of rewrites in places. As I've said before, this is not how I wanted to write the utility but I'm somewhat stuck with it now. It shows how much better things are when you do it properly.

Next time, we'll get down to the design and writing of a small (I hope) utility piece of code to make handling menus in your assembly language programs a bit easier and standardised. It's not going to be a world betare in programming excellence, but you'll see how I develop programs from start to finish.

See you next time.

EASYPTR and the New Window Manager

Dilwyn Jones

Now that people like Marcel Kilgus have put so much effort into enhancing the operating system, isn't it high time we software authors started to release programs to make proper use of not just the colours but also the facilities offered by the new window manager?

Of course, some already have. Most pointer driven programs are created with the combination of Easyptr and QLiberator. Up until now, it has only been possible to write high colour programs

using Easyptr by use of techniques such as those described by Wolfgang Uhlig in a recent QL Today.

TurboPTR and QPTR users have a bit more versatility and I will not be discussing those here. Very recently, Marcel Kilgus has put in some work on updating the Easymenu (menu designer) part of Easyptr. While he was something of a reluctant author when it came to Easyptr updates, nonetheless he did put in some work and at the time of writing it looked as though this would in time become an official update, as he has been given permission by the original author Albin Hessler to update the package.

Before the official update comes out, a small number of people have been issued with test copies of v4.00 of Easymenu, with occasional updates

beyond that. This version is not quite complete and although capable of producing working menus, there are gaps in what can be done and one or two ways to make it fall over. If you do successfully create menus and save them, you should be able to create working programs. When I originally wrote this article, there were limits, as publication grew nearer, more and more updates came and the package became more stable, although there was little by way of updated instructions.

The main thrust of this article is to show how to use the updated Easyptr to compose menus and write programs which use the System Palette. If you have read Wolfgang Lerner's descriptions of the new facilities in SMSQ/E 3.00 and later (Window Manager 2) you'll have noticed references to the various Window Manager colour handling formats.

There are one-byte colour values as before, palette values, grey scale (SMSQ/E prefers the American spelling of Gray) and a few others. These are fixed formats - you use a specified format and that's that. The system palette is a little different. With these (there are four possible palettes, but we'll pretend there's only one for the purpose of this article) we can assign colour lists to the menu elements which will use whatever colour scheme the user sets up via the Window Manager.

The system palette is a list of colours associated with the various elements of a menu. The reference you enter when specifying colours is an offset into the table of colours.

You will be pleased to know that once you have written a program to use the system palette there is little more you need to do to change the colours or handle the colours in any way. The user has control over the palette and when your menus have been programmed in the way I'll describe they will follow the user's choice of colours.

By using a program such as QCoCo, the user can create a new list of colours to reprogram the system palette and your programs can use these. The great beauty of this scheme is that all programs which use the system palette can look the same. If you fancy a change of colour scheme, you need only specify a new colour palette and all programs which use it will follow suit! No need to reconfigure all your programs. No need to cater separately for 256 colour and 16-bit colour modes unless you want to, although you may have to double up on menu definitions if you want your program to work on older QL systems as well.

When I first explored this facility after receiving a trial copy of Easymenu 4.00E from Marcel Kilgus, I was amazed at how simple it was to write programs to use the new system palette, so I thought I'd write up this description to help people who are already familiar with Easyptr to move up a step without the tears you often expect when moving up a programming level! Before I learned how to create programs which used the System Palette and all the new facilities, I was impressed by screen shots I'd seen and thought it would take me ages to learn to program the various colour schemes, 3D borders, sculpted looks of buttons and borders and so on. Once I'd got beyond first principles I realised it wasn't as hard as I'd assumed as long as I had Easymenu 4 to use.

What Do You Need?

You will need the following programs:

1. Easymenu 4.00 or better. You should configure Easymenu 4 with the MenuConfig program to set your preferred starting defaults, it only takes a moment or two.
2. A sprite editor if your menus need sprites. If you are happy with plain old mode 4 sprites, you can use Easysprite (which has been modified to work more reliably on the new systems, although it is still only able to design mode 4 or mode 8 sprites. If you want high colour sprites, you'll need to get a copy of Jérôme Grimbert's sprite editor.
3. A copy of a System Palette editor such as QCoCo (Wolfgang Uhlig - download it from his website at www.uhlich.nl/ql/ or get it from a PD library). This is only needed to help create different colour palettes to test how well your programs work with the new system. If you are happy to work blind, or already have example colour themes such as the one available on Marcel Kilgus's website, you don't even need QCoCo.
4. QLiberator, to compile the program (only if it has to be compiled). No special settings are needed, just use QLiberator to compile programs written with Easyptr menus in the normal way by appending the menus etc to the Easyptr extensions files such as easymen_cde or ptrmen_cde for example.

How Do I Get The New Colour Numbers Into Easymenu?

This is one of the major changes in the new Easymenu. In older versions, you could only get colour numbers 0 to 255 into menu items. Marcel

has now made it possible to enter the References in Table 1 as 4 digit hex values preceded by a \$ symbol (for hexadecimal) to distinguish them from the older colour numbers 0 to 255. What's more,

Marcel has kindly made the menu elements such as loose items and borders default to the relevant entries in the system palette unless you explicitly enter some other colour value.

Table 1 - System Palette items list

Name	Number	Meaning
sp.winbd	\$0200	Window border
sp.winbg	\$0201	Window background
sp.winfg	\$0202	Window foreground
sp.winmg	\$0203	Window middleground
sp.titlebg	\$0204	Title background
sp.titletextbg	\$0205	Title text background
sp.titlefg	\$0206	Title foreground
sp.litemhigh	\$0207	Loose item highlight
sp.litemavabg	\$0208	Loose item available background
sp.litemavafg	\$0209	Loose item available foreground
sp.litemselbg	\$020a	Loose item selected background
sp.litemselfg	\$020b	Loose item selected foreground
sp.litemunabg	\$020c	Loose item unavailable background
sp.litemunafg	\$020d	Loose item unavailable foreground
sp.infwinbd	\$020e	Information window border
sp.infwinbg	\$020f	Information window background
sp.infwinfg	\$0210	Information window foreground
sp.infwinmg	\$0211	Information window middleground
sp.subinfbd	\$0212	Subsidiary information window border
sp.subinfbg	\$0213	Subsidiary information window background
sp.subinffg	\$0214	Subsidiary information window foreground
sp.subinfmg	\$0215	Subsidiary information window middleground
sp.appbd	\$0216	Application window border
sp.appbg	\$0217	Application window background
sp.appfg	\$0218	Application window foreground
sp.appmg	\$0219	Application window middleground
sp.appihigh	\$021a	Application window item highlight
sp.appiavabg	\$021b	Application window item available background
sp.appiavafg	\$021c	Application window item available foreground
sp.appiselbg	\$021d	Application window item selected background
sp.appiselfg	\$021e	Application window item selected foreground
sp.appiunabg	\$021f	Application window item unavailable background
sp.appiunafg	\$0220	Application window item unavailable foreground
sp.scrbar	\$0221	Pan/scroll bar
sp.scrbarsec	\$0222	Pan/scroll bar section
sp.scrbararr	\$0223	Pan/scroll bar arrow
sp.buthigh	\$0224	Button highlight
sp.butbd	\$0225	Button border
sp.butbg	\$0226	Button background
sp.butfg	\$0227	Button foreground
sp.hintbd	\$0228	Hint border
sp.hintbg	\$0229	Hint background
sp.hintfg	\$022a	Hint foreground
sp.hintmg	\$022b	Hint middleground
sp.errbg	\$022c	Error message background
sp.errfg	\$022d	Error message foreground
sp.errmg	\$022e	Error message middleground
sp.shaded	\$022f	Shaded area
sp.3ddark	\$0230	Dark 3D border shade
sp.3dlight	\$0231	Light 3D border shade

```

sp.vertfill      $0232  Vertical area fill
sp.subtitbg     $0233  Subtitle background
sp.subtittxtbg $0234  Subtitle text background
sp.subtitfg     $0235  Subtitle foreground

sp.mindexbg     $0236  Menu index background
sp.mindexfg     $0237  Menu index foreground
sp.separator    $0238  Separator lines etc.

```

I have divided Table 1 with blank lines to identify the various groups of menu elements to make it easier to know what value to assign to loose items, borders, information windows, application window menus and so on. To appreciate what each bit is, you need to be familiar with the terminology. There are two excellent tutorials by our very own Norman Dunbar available from PD libraries - the PE Idiot's Guide and the Easypr Tutorial, both excellent get you going guides for their respective subjects.

Figure 1 shows the main command screen of the new Easymenu, so that we can be sure that you know which command I'm referring to! It also shows an easy way to check that you have a version of the new Easymenu - it shows version 4.00 or higher and Marcel Kilgus's name at the bottom of the screen.

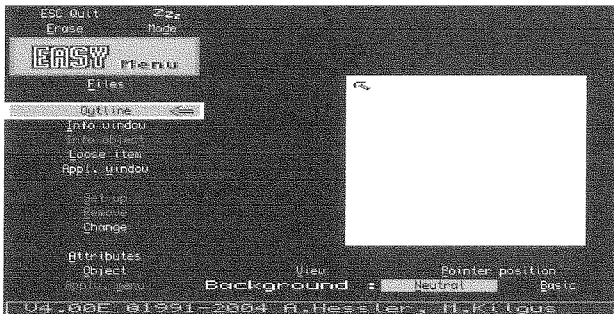


Figure 1 - Commands menu in Easymenu

The first step is usually to design the menu outline, unless the default happens to be exactly what you need. Using the Outline and Change commands in Easymenu, drag the corner out to the required size, add any loose items areas then select the Attributes command to specify the colours and border. See Figure 2 for the Outline Attributes menu. For those familiar with Easymenu it's very similar to older versions, but the stippled colours are replaced by the new colours when working in high colour modes, or replaced by the plain 4 basic colours when working in mode 4.

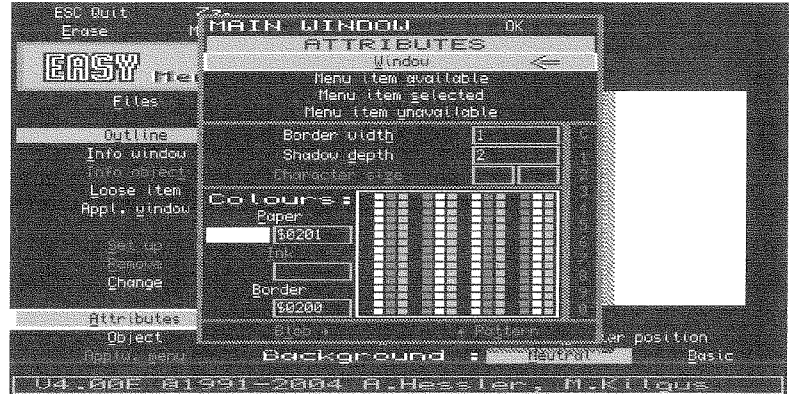


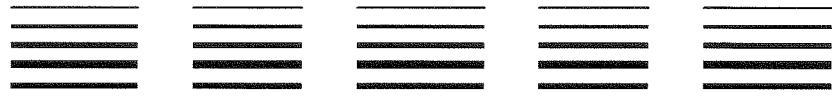
Figure 2 - Outline Attributes menu

What Easymenu calls the Outline is simply listed as Window in Table 1. The Window Background in Table 1 is what we have been used to calling Paper colour in Easymenu. The Window Border is the Border colour in this menu. The Ink colour in this menu would correspond to Window Foreground in Table 1, but Easymenu does not use this.

What Easymenu refers to as Menu Item Available, Menu Item Selected and Menu Item Unavailable are what Table 1 refers to as Loose Item Available, Selected and Unavailable respectively. This sort of thing (slight differences in terminology) needs to be kept in mind when working on these things and you quickly get used to it, especially if you have used QCoCo where all named items are fairly obvious.

Set the relevant colour Number for the 3 possible states of the loose items. Here also, Easymenu refers to Paper and Ink whereas table 1 refers to Background and Ink respectively. In the 'Available' section you also get to specify a Border value, which Table 1 refers to as Highlight, which is the box drawn around a loose item when the pointer is moved over it in a program. This highlight is item number \$0207 in the table and you'll see that Easymenu kindly offers this as default. Note that by default, Easymenu puts the relevant item colour number in the boxes for you. So in the Window Background (Paper) box, it places the value \$0201, the value in the NUMBER column in Table 1. The value \$0200 is placed in the Border colour box. So hopefully you can see how to use

QUANTA



Independent QL Users Group

World-wide Membership is by subscription only,
offering the following benefits:

Bimonthly Newsletter – up to 40 pages

Massive Software Library – All Free !

Free Helpline and Workshops

Regional Sub-Groups. One near you?

Advice on Software and Hardware problems

Subscription just £14 for UK members

Overseas subscription £17

Barclaycard: Visa: Access: MasterCard: Accepted

Now in our Twenty Second Year

Further details from the Membership Secretary

**John Gilpin, 181, Urmston Lane
Stretford, Manchester, M32 9EH (UK).**

Tel. +44 (0) 161 865 2872

or

Visit the Quanta Web Site

<http://www.quanta.org.uk>

E-mail: membership@quanta.org.uk

Let's CELEBRATE QL is 21

WORKSHOP & BIRTHDAY DINNER PARTY

at IBIS Hotel, Portsmouth, Hants.

Workshop including Demonstrations, Talks and Traders etc.

Ladies' Programme visiting Places of Interest

Saturday & Sunday October 29th & 30th 2005

Full information from Quanta Secretary

BOOK NOW!

these numbers - find the relevant entry in Table 1 for the element type you are specifying and use that. Most of the time Easymenu will suggest the right ones for you. Note also that Easymenu does not use or let you specify all possible elements. Some items such as Menu Index (the letters and numbers along the sides of a QSpread spreadsheet for example) are not used in Easymenu. Where more than one foreground colour is mentioned as being available for a menu element, Easymenu usually only uses the 'foreground' not 'middleground'. QCoCo refers to these as 'Ink 1' and 'Ink 2' for example. Although Easymenu doesn't let you use both directly, you could probably choose which of the two to use as 'Ink' in Easymenu, especially if there is more than one object in the same window to be displayed in different colours.

Although there is no 'Title' element as such in Easymenu, you could create an information window and information object and specify the Title colour numbers for these. The two background colours for a Title are essentially the equivalent of Paper (Title Background - the background of areas not including text) and Strip (Title Text Background - the area immediately behind text) in SBASIC.

Now that you are getting used to the idea of Easymenu offering you what is usually the right item Number as a colour number everywhere, you can use this to identify elements if you are unsure what a particular element is, due to the different terminology used in places.

What may not be so obvious is how to specify particular colours or border types. In fact, if you are opting to use the System Palette you do not really do this, you go by the colours defined in a system palette as selected by the user, not by you the programmer. You might want a particular title to be in red, and you could do that by simply selecting red from the list of colours in the Attributes menu in Easymenu or entering decimal number 2 as a value for colour. But if the user has a system palette colour scheme which happens to include red paper for a window, tough. I don't know about you, but I tend to find that red ink on red paper is sometimes a bit hard to read.

If you want to make your programs use the System Palettes, no fixed colours, OK? Let the user use his/her own colour scheme and specify the relevant \$xxxx number and play safe. Alternatively, if you want your programs to be one fixed colour, then go ahead and hard code the decimal colour number in as you did before. I have quickly grown to like the System Palette method because it is flexible. The user can use a default

system palette or his/her own preferred list of colours (and QCoCo is the perfect tool for creating this list of colours). The user changes the preferred colour scheme via the System Palette (only a few short lines in a simple SBASIC program and a list of colour numbers are needed to do this) and your program automatically follows suit, which is great for the user and saves you having to write complex colour configuration systems for each and every program you write, which is too much like hard work as far as I'm concerned.

Sometimes, we need to write something to a window by using the MWINDOW command to switch to a window, then we use INK and PAPER etc when printing text to a window. You can still 'hard code' your choice of colours for this if you wish to, but you run the risk of running into the 'same ink and paper' problem described above unless you introduce a configurable choice of colours of some sort for these 'hard coded' INK and PAPER commands. Do bear in mind that you can use WM_INK and WM_PAPER for example in place of INK and PAPER if you want to try to make things work via the Window Manager System Palette settings.

The Application Window items work in pretty well the same way as the menu outlines discussed above. An Application Window can have a menu displayed in it (the ones you set up with MAWDRAW or MAWSETUP) and the items in the menu can have colour pairs based on whether the item is Available, Selected, or Unavailable like a loose item. The application Window can have the usual Border (\$0216 in Table 1) and Paper/Background (\$0217), but not a Foreground/Middleground Ink which do not exist as user settings as such in Easymenu. Again, the Item Highlight (\$021A) is the little border drawn around the items in a menu list when the pointer passes over them.

The early version of Easymenu used for the purpose of this article seems not to have changed in terms of handling the pan/scroll bar and arrows, although there is a reference to such a change in the updates file of a new versions received just before this went to press.

I have never really used the Full Menu options in the Applw. Menu (Application Window Menu) command of Easymenu. I usually go with its default settings, specifying no menu here and set up the application window menus based on the arrays or array slices I pass to MAWDRAW in my programs to set up the menus at runtime. So I am as yet not too sure how to handle the Full Menu option here. The ONLY BARS option is of no use for SBASIC programmers anyway, it's for use by assembler

programmers. The Colours option in the ApplwMenu command seem to only allow you to specify colours 0 to 7 as they did before, I'm not sure if the Window Manager would allow other colours to be used or if it is better to stick with these 8 basic colours, or whether future versions of Easymenu will introduce any changes in this area.

How Do The Borders Work?

You do not really need to know how to specify border types as this is taken care of in the system palettes, but it can be handy to understand what's available, how they work, what's QL compatible and what isn't and so on in case someone asks why something is going wrong or why something is done the way it is, or simply if you wish to directly specify a border for a particular reason as opposed to using the default one.

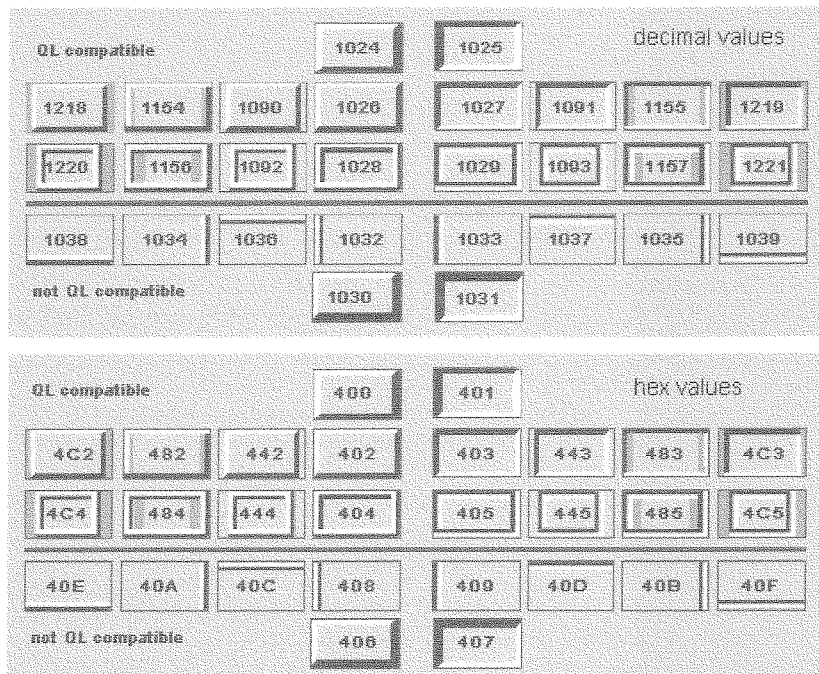


Figure 3 is a chart designed by Marcel Kilgus which I have "stolen" from Wolfgang Uhlig's website to help illustrate the borders. This chart is useful for checking how a border is meant to look and what's QL compatible and what isn't. It also helps you to see how the 3D effects are built up, which way the shadow of the sculpting effect works (what looks raised and what looks lowered and so on). The top half gives the border 'colour' values in decimal, while the bottom one is in hexadecimal. Above the line, the borders are described as 'QL Compatible' and below the line as 'not QL compatible'. Of course, if it is a GD2 program which doesn't have to rely on QL compatibility, use any of them. It might not be so

obvious on paper, but you will probably grasp the notion that since the light source is meant to be top left of a border, a border looks 'raised' if the light edge is above and left, and looks recessed if the light shade is bottom and right. When you see them on screen in a program, border number 1024 around a rectangle is the effect of a simple raised rectangle and 1025 is a pressed down version of it. Border 1028 is slightly different for example - it is the border itself which looks raised. This might look good as the border around an information window or application window for example, but may not be much use as a 'highlight' (where you move the pointer over a loose item or application menu item and a border appears). Some of these borders may use more pixels than traditional borders and reduce the space available in a loose item slightly. Since loose items and application window menu items are often exactly

one line of text high, or the exact size of a sprite, some care and thought is needed where you plan to use some of the 3D borders. After a while, you get used to it, but if you use the System Palette borders and the user has set up a thick border for a particular item type and your program runs into out of range errors for example, it is something to suspect - it may be a classic case of something the user has set up in a System Palette which your program has little control over. Ultimately, users set up the system palette colours list and you only tell Easymenu which of the Window Manager options to use.

I'd Like My Program To Run In Both QL And High Colour Modes.

Now we come to the less easy bit. There are two possible approaches, both of which may need two different sets of menus for the different colour modes.

In your program's configuration, you could offer the user a choice of running in Mode 4 colours or High-colours, or an Automatic mode if you are confident you can rely on your automatic mode for detecting the required colour mode and window manager version availability. Although I haven't really got this far yet in my programming, I foresee my programs having an item in the config block such as this:

- Run in 1. QL colour mode all the time
- 2. High Colour mode all the time
- 3. Auto-detect which mode to use

Option 1 just uses QL mode menus all the time, useful if a program is used often on both a modern Window Manager 2 system and an older QL system without Window Manager 2. 100% compatibility, but no bells and whistles and fancy displays!

Option 2 just tries to run the program in high co-

lour mode all the time. If the program finds itself running in QL colour mode it may well fall over with an error message (actually, it almost certainly would, not just "may"). This is only useful where the program is only ever used on this type of system.

Option 3 tries to give the best of all worlds. The trouble is how to do the necessary detective work in your programs.

Examples of useful lines of code can include:

```
REMark assume we can't use the System Palette menu versions
use_syspal_menus = 0
IF VER$ = 'HBA' THEN
  REMark it's an SMSQ/E system if it has HBA version of BASIC
  IF VER$(1) >= '3.00' THEN
    REMark SMSQ/E 3.00 on QPC2 was the first with Window Manager 2
    IF DISP_TYPE >= 2 THEN
      REMark the computer is running in 256 colour or 16 bit mode
      REMark set a flag value here to indicate which set of
      REMark menus you intend to draw in your program
      use_syspal_menus = 1
    END IF
  END IF
END IF
```

DISP_TYPE is an SBASIC extension which checks the current display type - 2 is a 256 colour mode of Aurora or QPC2, 3 is a 16-bit colour mode, future systems may feature higher colours I suppose, hence the need for the test '>= 2' rather than explicitly testing for the values 2 and 3. VER\$(1) returns the operating system version number.

The Window Manager and Pointer Interface have their own version numbers. As far as I know, there

is no function to test and return these numbers, but if you have the Display_cde extensions from my article in Volume 7 Issue 6 of QL Today (extensions available from my website) that set of extensions includes one called WMAVER\$ which tests the Window Manager version number for a given channel number. It also includes extensions to test for the presence of pointer environment and window manager:

```
IF PTR_ENV (#channel) = 0 THEN
  PRINT'Sorry, no pointer environment.'
  INPUT'Press ENTER ';z$
  STOP
END IF
IF WIN_MAN (#channel) = 0 THEN
  PRINT'Sorry, no Window Manager.'
  INPUT'Press ENTER ';z$
END IF
use_syspal_menus = 1 : REMark assume we can use new menus
IF WMAVER$(#channel) < '2.00' THEN
  PRINT'Window Manager too old for posh new menus.'
  use_syspal_menus = 0 : REMark oh no we can't use new menus!
END IF
```

There's also a function to test for GD2 (the "colour drivers" or Graphic Device 2) if you want to test everything:

```
IF GD2(#channel) = 0 THEN PRINT'Oops, no GD2!'
```

As with anything else in this article, a lot of this is new to me so if you spot errors or bad practice in the above, please let me know and I'll print corrections in a future issue.

I Haven't Heard Of Some Of The Items In Table 1

OK, let's try to split it into groups by function:

Window - the main menu outline. Everything else is contained within this.

Title - not directly used in Easymenu, the titles you often see in the top line of a menu in some programs, e.g. the menus in QMenu. Possible to

put a short text as an information object inside an information window and specify the \$0204, \$0205 and \$0206 numbers for this if you wish, although probably not best practice.

Loose Item - As you would expect. Clickable items within a menu, which can be positioned just about anywhere, hence the name 'loose item', as they are often one off items, not necessarily part of a menu in a regular list format.

Information Window - As you would expect, just simple text items which are not selected by clicking on them. In Window Manager 2, there are two ink colours possible, the foreground and middleground (the term middleground used as distinct from background, the second ink is somewhere between the foreground and background colours!

Subsidiary information windows - not directly used in Easymenu, although you can define an information window and give it the attribute number for Subsidiary windows to make use of them, although probably not best practice. In some respects, these could be called Secondary Information Windows, but this risks confusion with the term Secondary Windows which has another meaning in pointer environment (the lowest window channel number is usually the 'primary channel', often, but not always, channel #0, and secondary windows are further windows defined within the outline of the main or 'primary' menu's channel, and at that point I'll beat a retreat before I get into deep water)

Application Window - As you'd expect. A general purpose window which can contain a menu, pan and scroll bars and so on. You can use MWINDOW in Easyptr extensions to switch a window channel to this window and use it to display text, user prompts, ask for INPUT etc pretty much as you can with most menu elements such as loose items and information windows. Application Windows are a little more advanced in that they can contain complete selection lists called Application Window Menus, such as lists of filenames and so on.

Pan/Scroll Bar - usually displayed on the right or bottom of a window where there are more items to display than can fit in the window, e.g. a long list of files. They usually show a bar within a longer strip which represents the visible part of a list in relation to the whole list, hence the names 'scroll bar' and 'scroll bar section'

Pan/Scroll Bar Arrow - indicates the menu can be scrolled in that direction if there's more than can be displayed. Clicking on these arrows usually pans or scrolls the menu in that direction.

Button - When a program is minimised into a

button (a little square representation of a program usually in a strip at the top of the screen if you have the QPAC2 button frame present) you can specify the attributes of the button with this setting. Not directly specified within Easymenu.

Hint - A little window which pops up if you hover the pointer over a command in some programs like Jochen Merz's QD. Not many programs support Hints at the moment. Not directly used in Easymenu, but you could achieve something similar with a menu outline set to these attributes and when testing for pointer activity with MCALL or MCALLT you could probably do something like IF no activity THEN Display_Hint which examines the pointer position and calls a routine to display a relevant hint window which is cleared if hit or if activity resumes (I don't really know, I've never tried programming hints or bubble helps as they're sometimes better known as in some applications. Hints can have two ink colours.

Error - used for Error message displays. Can be a window or menu set to use item numbers \$022C to \$022E

Shaded area - used for shading parts of a border for example.

Dark and Light 3D border shade - used for the sculpted borders, the light section generally faces the light source and the dark section is the shaded bit away from the light.

Vertical area fill - used in some of the 3D borders.

Subtitle - like the Title items, but a secondary (i.e. not main) title.

Menu index background, foreground and separator lines etc. These are used for grids and index strips around spreadsheets, for example. QSpread is currently the only program I am aware of which uses this.

What Do The New Menu Layouts Look Like

As QL Today is not a colour magazine we cannot simply print a picture of a program and show you the difference that the System Palette choices of colour schemes can achieve, but I'll include a couple of diagrams and suggest what to look out for when you see a real, live program using the System Palette.

Figure 4 shows two copies of my first such program running side by side in both standard QL colour scheme and also running with System Palette colour scheme menus. Hopefully, you may be able to see differences even in black and white in this magazine. The picture on the left is the new version, whereas the one on the right uses the

old mode 4 colour schemes I used in programs like Launchpad, which was basically red icons on a white background and simple red or green borders or highlights. The one on the left is displayed as shades of grey. The application window at the bottom (the one with the list of filenames) is meant to look slightly lowered or recessed compared with the areas around it because the border around the edges of the files list box has a shaded effect to make it look slightly 3D. When the pointer is moved over either a filename or the drive/directory name box, a 'raised' border appears around it, looking like a button being switched upward, which on a colour display at least looks much nicer and more professional than the simple 'black box' where the drive name is shown in the picture on the right. Also, if you look at the scroll bar on the right, you'll see that the QL version is a simple set of red lines (OK, this is a black and white picture!) on a white background whereas the one on the left has a generally 'sculpted' look which is a bit easier on the eye than the harsher mode 4 version. I used the same mode 4 sprites in both programs for simplicity but I could have used J. Grimbert's sprite editor to create more professional looking loose item icons for the high colour versions of the menu if I wanted to.

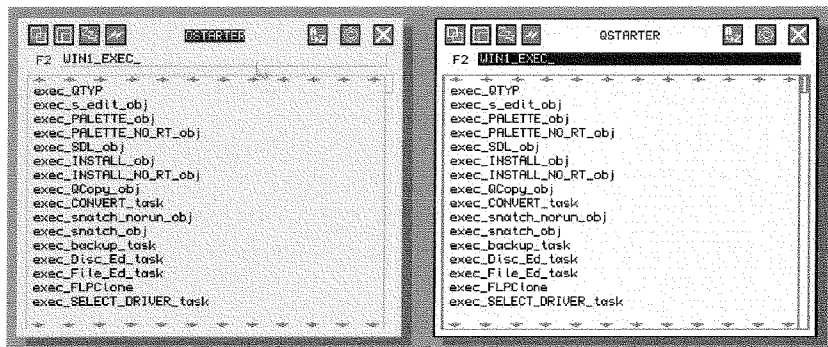


Figure 4 - Comparison of how the same program can look under both old and new systems

Figure 5 is also worth a look, as it shows effects which can be produced with information windows. Here you see a copyright notice screen divided into sculpted sections for a nicer more professional look, more in line with modern computer software practice. In this, the title at the top looks recessed or countersunk, as are my email and postal address. The little dragon logo I use in the copyright screens in my programs is also shown recessed so you can see how text boxes and icon pictures might compare. The highlight which appears around loose items (the 'OK' in this screen dump) goes the opposite way, the bevelled edge looks raised when the pointer

passes over it, a nice contrasting effect which might look a little odd at first when you are used to simple flat traditional QL graphics, but something you quickly get used to and which most people like and prefer to the original QL graphics.

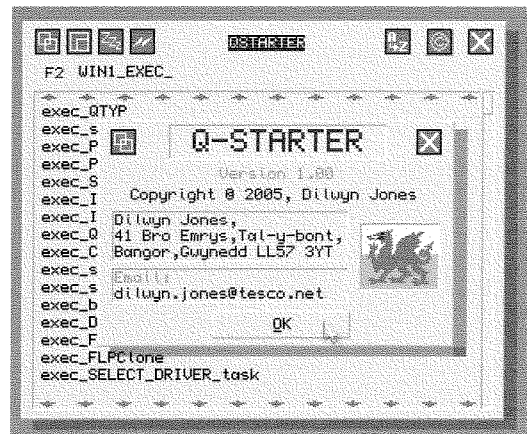


Figure 5 - Another example '3D' menu

Summary

This article is meant to be a summary of my ventures into the System Palette and how to use EasyMenu to create programs capable of taking advantage of the facilities provided by the colour drivers and Window Manager 2 in particular.

The article is based on my short experience of writing such programs, reading articles by people like Wolfgang Lernerz and Marcel Kilgus, and using the excellent QCoCo program without which I'm afraid Window Manager 2 might have passed me by or taken longer to get used to. I hope this article has shown that polished professional programs can be produced to use facilities of the new Window

Manager quite easily with tools like Easypr. I do not claim to be an expert on the subject, indeed the "experts" may well see bigger holes in this article than the English rugby team defence in the current 6-Nations championship (I can probably safely say that this year as Wales have finally won it after all these years in the rugby wilderness!). I welcome feedback, corrections and suggestions about this article from more "expert" people in this field. I will endeavour to publish any corrections and feedback in future issues of QL Today. Even if this article is found to be full of errors, I do hope it will go some way towards helping software authors to start producing ever better QL software to make better use of the hard work of the operating system designers.

Updates at/from J-M-S

Jochen Merz

First, the office situation: the office has partially moved now - not into the room which is planned to become the main office room, more into the storage room. Anyway, it is set up and working, I sit here in front of my laptop and prepare QL Today on it. No photos this time - apart from the fact that we had to scratch off the ceiling, prepare it and paint it again there's only a proper floor now.

I really hope to be able to create the next QL Today in my main office on my main PC ... we'll see, I'll keep you up-to-date.

Phone and fax are working, there was a short two-day switchover period end of March ... so if you tried to fax your renewal and it did not get through, then please retry.

As announced before the move, the J-M-S BBS had to be switched off. Not only the phone number has to be used for something else, but the internet is a much better, more flexible and cheaper method of updating software.

My call for help about a year ago was heard and Bruce Nicholls has programmed a wonderful tool which he runs on his server - a website for handling updates! Thanks a lot Bruce, it is great!

The URL is easy to remember:
updates.j-m-s.com

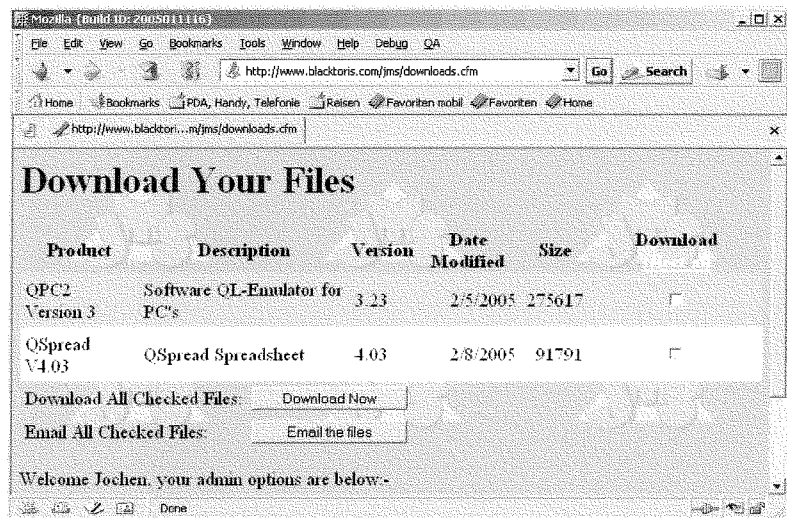
and it will redirect you to Bruce's server, as shown in the screenshot.

At present, we're running tests, and there are not many files on the update server. But it will change soon.

A brandnew QSpread is already on the server.

If you want to be registered for server access, then please send an email to **SMSQ@J-M-S.com** and provide me with the following information:

- Your name
- Your J-M-S customer number (if known)
- Your login name for the server (choose one, no spaces and special characters please)
- Your password for the login (choose one)
- and a list of the programs from J-M-S which you own.



The setup may take a short while and I'll inform you as soon as your download account has been created.

Please do not send mails before end of April, as I still need some time to set it up properly and I am busy preparing QL Today currently - as you probably can imagine.

Interpolating Curves - Vector Graphics

Stephen Poole

In 1984, when I was working as a Land Drainage Surveyor, I decided to use my QL to computerise all the trigonometrical and hydrogeological calculations and automate the production of plans and projects.

This turned out to be quite straightforward until I hit the thorny problem of drawing contours. Indeed, most draughtsmen at that time preferred to draw contours by hand, a time-taking task requi-

ring considerable skill, (including isobars on those weather charts on TV!). So before you read this article, ask yourself the following questions that have been plaguing mathematicians for centuries: How do you define a curve? In what way can a variable curve be approximated?

Reading French computer magazines at the time, I came across various programs that attempted to reply to this,

including the now-famous Bezier-curves. These Bezier constructions are simple to program but cut off corners. This was inadequate for contours, which must pass through end-points smoothly. So I began by experimenting with various ideas, starting by breaking curves down into sections of circles and then stringing them together. This worked well except for tight turns where the joints were unacceptable. But that program laid the groundwork for 'sorting' the general orientations of subsequent sections of curves, selections based on fairly standard surveying principles. Basically, each section of a curve is tested to determine its orientations and treated accordingly. After much thought, I realised that I could generate almost any curvature using perspective projections of ellipses drawn as inclined planes... i.e. conic sections.

But first I needed to devise my own elliptic arcs routine to improve on the QL's ARC and ELLIPSE keywords, neither of which could draw elliptic arcs. This proved to be not too difficult, but there still remained the problem of the projections. After comparing points three by three, the challenge was thus how to link them with elliptic arcs. The solution simply confines the curves within triangles formed by subsequent pairs of concentric tangents. Then using the Cosine rule (with a ponderation factor 'Fc' to determine the degree of ellipticity), thereby determining the curve's intermediary 'projected' points by the use of simultaneous equations. Oof!

So that was the theory, tested by hand, and all that remained was to write all the code. Luckily I had already written routines to resolve simultaneous equations and suchlike,

and after assembling the various component parts I had my first 'contours' program ready for debugging, and on the whole, it worked as expected. My first remark was that the program needed an awful lot of code which necessarily slowed it down in comparison to Bezier curves, but also that the curves produced could be interpolated as smoothly as necessary by linking as many intermediate points as required, with short straight lines. When 'anchor-points' are close to each other, curves may seem a little 'tight', but that is the way they should be, (as are curves drawn with stencils under the same conditions). Curves should be rendered better, that is, with less aliasing, on more recent monitors than mine (with its original 512x256 definition). It would be interesting to see if aliasing is reduced by directing output to a good plotter.

By the time I had written the first draughts of this program, (1985), I had left my surveying job to go to the Institute for the Promotion of Industry at Colmar and study Industrial Ecology, and wondered what other uses the program could possibly have. At that time, computer magazines were full of talk of Vector Graphics and it occurred to me that my unused 'contours program' could be adapted to compress complicated surfaces, by selecting just a few points on their perimeters and letting the contours program interpolate and reconstruct the entire shape. After some minor modification, that program proved to be effective too, and so I wrote an additional short routine to draw Chris Boardman in full effort breaking the world indoor cycling speed record. At that time, the program was sent in to the Quanta Library as I thought it might have potential

for more development, so I made it available to Quanta members. Then Line Design appeared, providing convenient Vector Graphics on the QL, so I stopped work on my program and moved on to other things. The contours program is somewhat inelegant, as at the time I wrote it I was short of microcassette space and so I optimised the code for length at the expense of clarity. At the time this facilitated editing, but nowadays I wish I still had the long-hand commented originals which I cannot access, as those cartridges are no longer readable! (Thank heavens I have copied all viable cartridges onto floppies... and will copy those too before their life-span exceeds the time-limit).

There is very little code which the user can usefully modify, as most of the variables are essential and must use set values. However, try changing 'num' to increase or decrease the smoothness of the curve. 'Num' must be a multiple of 4 so that the elliptic-arcs routine functions correctly. Try also altering the curvature-factor 'fc'. This program is essentially a demonstration of the original test-routine which used random values to try out all configurations. If I were to use it professionally, I would improve the case where a curve joins a straight line, and also provide a U-turn. Unfortunately I do not have the necessary spare time to do a complete rewrite to make the subroutines more explicit, unless readers ask me specifically. (The program is printed here particularly as a demonstration of an unpublished method. I would have to change the Initialisation routine to make it freely usable in other applications). I suspect that there are much faster programs for PCs, bearing in

mind the speed with which Vector-Fonts are printed on screen, (using dedicated graphics cards), but as such programs go unpublished, I have no idea how they work.

On a different tack, the whole problem of FILLing imbricated curved surfaces depends on how much care you take when you define the anchor-points initially. (Or you need dedicated FILLing programs, but that is quite another subject). Perhaps readers who know of other interpolation programs may wish to share their knowledge of them with us all. As soon as I get enough time I will trawl the Web looking for curve-fitting examples. If you have any ideas for adapting the program for other uses please feel free, as QL Today readers, to do so. Or send me an email at the above address if you wish to have any further details. (My 'Minitel' email terminal is 100% virus free, so I don't mind having its address printed). Remember, interpolated curves are by definition approximations, but should be adequate

as guidelines, (bearing in mind that contours were previously guessed at when drawn by hand..). Jochen has a copy of the 'Chris Boardman' program 'bike_bas', so contact him if you wish to see Vector Graphics in action on the QL. I do not intend to publish the bike program as it uses 95% of this present program's code. I have also sent Jochen my original 'QDOS_conics' program, which (mis-)used the MlStake keyword to allow full-page searching and editing of all procedures by a D.I.Y menu system. Of course, SMSQ/E does not allow this, so I now do the same thing using linked GOTO's, and which may form the basis of a subsequent article. Perhaps Dilwyn will put these programs on his website as they all require a lot of typing in, unless you use Optical Character Recognition scanning, as does Ian Pizer. Lengthy code is also generally the case for Herb Schaaf's 'Gee Graphics' series. (Remember that 'bike_bas' works through SuperBasic initialisation. That means that all changes in out-

put demand modifications to the INITialisation code, as there is no front-end interface). I have barely mentioned Vector-Compression routines yet, which seem to be all the rage at the moment. But be aware that the bike program uses relatively few well-spaced out coordinates to define its curved surfaces, as compared to generally massive bitmap requirements. This also introduces us to automatic profile-recognition techniques, but that too will have to be the subject of another article.

This interpolation program was one of the most challenging programming projects I ever indulged in...and one of the few I have barely used subsequently. Alas! as it would be in a far better state now! But that being said, if I had continued in my surveying job, this would have been one of the most highly used programs in my suite. The main difficulty was in conceiving a method that would work in every case. Oh Dear! I hope I found all the bugs!

stevepoole@minitel.fr

```

100 ::
110 REMark conics_bas, CURVE_INTERPOLATION by S.Poole. v15nov2004
120 REMark When done, hit the 'q' key to quit.
130 CLEAR
140 :
150 OPEN#1,con_128: WINDOW 512,256,0,0: SCALE 250,-70,-50
160 PAPER 0: INK 4: CLS
170 sg$='++ +0 +- 0+ 00 0- -+ -0 -- ': sig$='-0+'
180 x=1: y=2: azt=3: trn=4: typ=5: fc=.7071068
190 Ph=PI/2: Pl=Ph*3: Pw=PI*2: ee=1: pp=2: num=32
200 RESTORE : READ dots
210 DIM t(dots+4,typ),L(3,4,2),u(pp,37,y)
220 :
230 REMark Initialise the end-points randomly:
240 FOR f=1 TO dots: READ tfx: t(f,x)=tfx+RND(40)-RND(80)
250 FOR f=1 TO dots: READ tfy: t(f,y)=tfy+RND(40)-RND(80)
260 FOR f=2 TO dots
270     IF t(f,x)=t(f-1,x): t(f,x)=t(f,x)+1
280 END FOR f
290 FOR f=2 TO dots
300     CIRCLE t(f,x),t(f,y),2: CURSOR t(f,x),t(f,y),0,0: STRIP 2: PRINT f: STRIP 0
310     IF t(f,y)=t(f-1,y): t(f,y)=t(f,y)+1
320 END FOR f
330 FOR f=1 TO 4: t(f+dots,x)=t(f,x): t(f+dots,y)=t(f,y)
340 :
350 DATA 19, 43,65,90,108,124,132,139,136,157,173,183,209

```

```

360 DATA 220,151,86,33,36,32,13,83,92,80,72,64,81,94,120
370 DATA 113,108,82,74,133,168,159,143,123,102,77
380 CIRCLE t(1,x),t(1,y),2
390 :
400 REMark Join the end-points with straight lines:
410 FOR f=1 TO dots+3
420   j=f+1: xx=t(j,x)-t(f,x): yy=t(j,y)-t(f,y)
430   dd=SQRT(xx*xx+yy*yy)
440   sx=(xx>0)-(xx<0): sy=(yy>0)-(yy<0): az=0
450   IF dd: az=ASIN(yy/dd)
460   SElect ON sx
470     =1: IF sy=-1: az=Pw+az
480     =REMAINDER : az=PI-az
490   END SElect
500   t(f,azt)=az: LINE TO t(f,x),t(f,y)
510 END FOR f
520 :
530 REMark Calculate the line-bearings:
540 FOR f=2 TO dots+3
550   th=t(f,azt)-t(f-1,azt)
560   IF th<-PI: th=th+Pw: END IF : IF th>PI: th=th-Pw
570   t(f,trn)=th
580 END FOR f: t(1,trn)=t(dots+1,trn)
590 :
600 REMark Determine the angles between successive lines:
610 FOR f=1 TO dots+3
620   t1=t(f,trn): t2=t(f+1,trn)
630   s1=(t1>0)-(t1<0): s2=(t2>0)-(t2<0)
640   s$=sig$(s1+2)&sig$(s2+2)&' ': tp=s$ INSTR sg$
650   t(f,typ)=((tp+2)/3)-1
660 END FOR f: t(dots+4,typ)=t(4,typ): INK 7
670 :
680 REMark Ascertain the bend type:
690 FOR f=3 TO dots+2
700   a=f-1: b=a+1: c=b+1: tipes=t(f,typ)
710   bx=t(b,x): by=t(b,y): cx=t(c,x): cy=t(c,y)
720   BCx=bx-cx: BCy=by-cy: MCx=BCx/2: MCy=BCy/2
730   Mx=bx-MCx: My=by-MCy: ab=t(b,azt)
740   a1=(t(a,azt)+ab)/2: a2=(ab+t(c,azt))/2: a3=ab+Ph
750   :
760   SElect ON tipes
770     =0,8: L(1,1,x)=bx+COS(a1): L(1,1,y)=by+SIN(a1)
780     L(1,4,x)=cx+COS(a2): L(1,4,y)=cy+SIN(a2)
790     L(1,2,x)=bx: L(1,2,y)=by: L(1,3,x)=cx: L(1,3,y)=cy
800   END SElect
810   :
820   SElect ON tipes
830     =1,2,6,7: L(2,1,x)=bx+COS(a1): L(2,1,y)=by+SIN(a1)
840     L(2,4,x)=Mx+COS(a3): L(2,4,y)=My+SIN(a3)
850     L(2,2,x)=bx: L(2,2,y)=by: L(2,3,x)=Mx: L(2,3,y)=My
860   END SElect
870   :
880   SElect ON tipes
890     =2,3,5,6: L(3,1,x)=Mx+COS(a3): L(3,1,y)=My+SIN(a3)
900     L(3,4,x)=cx+COS(a2): L(3,4,y)=cy+SIN(a2)
910     L(3,2,x)=Mx: L(3,2,y)=My: L(3,3,x)=cx: L(3,3,y)=cy
920   END SElect
930   :
940   SElect ON tipes
950     =0: TRI 1,0: =1: TRI 2,0: =2: TRI 2,0: TRI 3,1
960     =3: TRI 3,0: =5: TRI 3,1: =6: TRI 2,1: TRI 3,0
970     =7: TRI 2,1: =8: TRI 1,1
980   END SElect
990 END FOR f: LINE TO cx,cy: i$=INKEY$(#1,333)

```

```

1000 IF i$=='q': STOP: ELSE GO TO 160
1010 :
1020 DEFine PROCedure TRI(TR,CW)
1030 LOCAl f,j,ct: REMark Simultaneous equations:
1040 xa=L(TR,1,x): ya=L(TR,1,y): xb=L(TR,2,x): yb=L(TR,2,y)
1050 xc=L(TR,3,x): yc=L(TR,3,y)
1060 xd=L(TR,4,x): yd=L(TR,4,y): BCx=xb-xc: BCy=yb-yc
1070 MCx=BCx/2: MCy=BCy/2: Mx=xb-MCx
1080 px=xa-xb: py=ya-yb: qx=xc-xd: qy=yc-yd
1090 g1=0: g2=0: IF px: g1=py/px
1100 IF qx: g2=qy/qx
1110 k1=ya-xa*g1: k2=yc-xc*g2: gg=g1-g2: My=yb-MCy
1120 ix=0: IF gg: ix=(k2-k1)/gg
1130 iy=g1*ix+k1: Maj=SQRT(MCx*MCx+MCy*MCy)
1140 IF px=0: ix=xa: iy=yc+g2*(xa-xc)
1150 IF qx=0: ix=xc: iy=ya+g1*(xc-xa)
1160 :
1170 IF TR=2 THEN
1180     ix=(ix+bx)/2: iy=(iy+by)/2
1190     ELSE
1200     IF TR=3: ix=(ix+cx)/2: iy=(iy+cy)/2: END IF
1210 END IF
1220 :
1230 mix=ix-Mx: miy=iy-My: im=SQRT(mix*mix+miy*miy)
1240 Min=Maj*fc: lim=2*Min
1250 bm=SQRT(BCx*BCx+BCy*BCy)/2
1260 ibx=ix-xb: iby=iy-yb: ib=SQRT(ibx*ibx+iby*iby)
1270 dv=(bm*bm+ib*ib-im*im)/(2*ib*bm)
1280 hi=ib*SIN(ACOS(dv)): ec=hi/lim*1: s1=TR
1290 :
1300 SElect ON s1
1310     =1: xE=Mx: yE=My: =2: xE=bx: yE=by: =3: xE=cx: yE=cy
1320 END SElect : so=hi>lim
1330 :
1340 IF so THEN
1350     ek=lim/hi: EXx=ek*mix: Ey=ek*miy
1360     ix=xE+EXx: iy=yE+Ey: ec=fc
1370 END IF
1380 :
1390 Min=Min*ec: ecc=Min/Maj: ro=0: h=num
1400 IF xc<xb: ro=180: END IF : q=h+1: st=360/h
1410 rot=0: IF MCx: rot=DEG(ATAN(MCy/MCx))
1420 rot=ROT(rot+ro): h=h-4
1430 hh=h/2: ct=0: SS=-st: AA=180-st: ZZ=st
1440 IF CW=0: SS=st: AA=-AA: ZZ=-ZZ: ct=hh
1450 :
1460 FOR f=AA TO ZZ STEP SS
1470     IF f=90 OR f=-90 THEN
1480         NEXT f
1490         ELSE
1500             ct=ct+1: j=ROT(f): Crx=Maj*COS(j)
1510             END IF : ROTATE pp,Crx,0
1520             cry=Maj*SIN(j)*ecc: ROTATE ee,Crx,cry
1530 END FOR f
1540 IF CW: j=0: ct=hh: ELSE : j=hh: ct=h
1550 x1=ix: y1=iy
1560 :
1570 FOR f=j+1 TO j+hh
1580     x2=u(2,f,x): y2=u(2,f,y): x3=u(1,f,x)
1590     y3=u(1,f,y): x4=u(1,ct,x): y4=u(1,ct,y)
1600     xp=x1-x2: yp=y1-y2: xq=x3-x4: yq=y3-y4: g1=0
1610     IF xp: g1=yp/xp: END IF : g2=0
1620     IF xq: g2=yq/xq
1630     k1=y1-x1*g1: gg=g1-g2: m=0

```

```

1640      IF gg: m=(y3-x3*g2-k1)/gg
1650      n=g1*m+k1: IF xp=0: m=x1: n=y3+g2*(x1-x3)
1660      IF xq=0:m=x3: n=y1+g1*(x3-x1)
1670      LINE TO m,n: ct=ct-1
1680 END FOR f
1690 END DEFine
1700 :
1710 DEFine PROCedure ROTATE(nm,rx,ry)
1720  LOCal rh,a,b,c,d,h,n
1730  c=(rx>0)-(rx<0): d=(ry>0)-(ry<0)
1740  n=-1: h=SQRT(rx*rx+ry*ry): a=ABS(ASIN(ry/h))
1750  SELEct ON c
1760    =-1: SELEct ON d: =-1: a=PI+a: =0: a=PI: =+1: a=PI-a
1770    = 0: SELEct ON d: =-1: a=P1 : =0: a=0 : =+1: a=Ph
1780    =+1: SELEct ON d: =-1: a=Pw-a: =0: a=0 : =+1: a=a
1790 END SELEct
1800 :
1810 b=a+rot: c=1: d=1: xx=0: yy=0
1820 REPEat loop: IF b<0: b=b+Pw: ELSE EXIT loop
1830 :
1840 IF b>=Pw: b=b-Pw: GO TO 1840
1850 IF (b==0) OR (b==Pw): xx=h: GO TO 1940
1860 IF b==Ph: yy=h : GO TO 1940
1870 IF b==PI: xx=-h : GO TO 1940
1880 IF b==P1: yy=-h: GO TO 1940
1890 IF b>P1 : b=Pw-b : d=n: GO TO 1930
1900 IF b>PI : b=b-PI: c=n:d=n: GO TO 1930
1910 IF b>PI: b=PI-b: c=n: GO TO 1930
1920 :
1930 xx=h*COS(b)*c: yy=h*SIN(b)*d
1940 u(nm,ct,x)=Mx+xx: u(nm,ct,y)=My+yy
1950 END DEFine
1960 ::

```

Editor's comment: as Stephen mentions biker_bas in this article, I'll print it in the next issue - the result looks good. As the current issue has already 60 pages (wow! only very few ads), there is no room.

Thanks also for the compliments, Steve, and as you can see, Herb Schaaf is back with another part of Gee Graphics in this issue.

Al Boehm

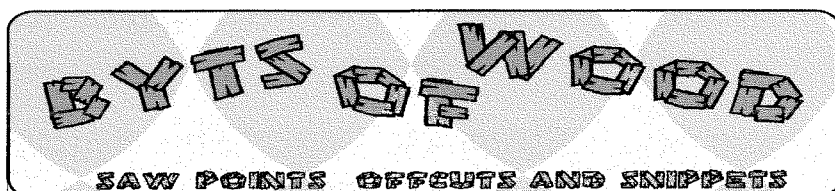
It was with great sadness that we heard from Al Boehm's daughter Lisa, via Bill Cable, of the illness of Al Boehm. We understand that he has suffered a stroke while visiting his daughter in California early in April, although the latest details we had at the time of writing were that he was joking with his family in hospital and was even able to remember and hand over his email password to Lisa.

For a long time, Al has been a prominent member of the QL community in North America and was closely involved with NASQLUG user groups and the annual USA QL show. We would like to offer Al our best wishes and hopes for a full and speedy recovery from his illness.

Jochen Merz, Marcel Kilgus, Tony Firshman and Roy Wood, the European contingent who have travelled to the US QL shows for the last few years, would like to extend their wishes for the speedy recovery of Al Boehm after hearing the news of his recent illness. He has long been one of the most active of the US group and one of the few to make the journey in this direction to attend the shows here. Without him many of the US shows would not have happened and he has been the driving force behind many other QL related projects. We understand, from his daughter Lisa, that he is making a good recovery and is already joking with his carers. Given his positive, outgoing, outlook on life and the determination he has always shown in everything I am sure he will be back on his feet soon.

We wish him well and look forward to seeing him soon.

Last Minute News: Al Boehm had a successful carotid artery surgery. He is on the road to recovery for some left side paralysis. He is now able to eat food again and is regaining strength every day.



One thing which amused me a lot after the QDT demo hit the streets was the number of experienced QL people who completely ignored the instructions on how to install it. There were quite a few people reporting errors which were just a result of trying to install it without letting it get on with it and install itself. I suppose that is because we are not very used to this kind of thing for the QL. Those of us who consider ourselves experts and therefore requiring no instructions all fell at the first post. I must admit that, had I not been running specific tests for Jim I may well have done the same thing so I am not poking fun here.

A Quiet Revolution

As I said in a previous issue, using QDT has revolutionised the way I use my system. I am hesitant about saying things like that in this column because it may be construed as free advertising since QBranch is one of the distributors but I do have some specific points to make.

One thing I have now done is to remove several lines from my BOOT file. I used to have a series of HOTKEYS set up to call versions of QD configured with different help files, usage settings and 'F10 Things'. This meant that I could call a QD which used the QBASIC 'Thing' and had Rich Mellor's QHelp system set up for writing a bit of BASIC that I wanted to compile with QLiberator and a completely different version of QD when I wanted to edit my BOOT file or even write this column. Now I have all of those

set up in QDT with slightly modified icons and five lines have gone from my boot file. And this is just the top bit of the cold floaty thing.

One of the things I did find threw me in a state of complete confusion. I had never realised something about QD until I started to set the different programs up like this. After I had been working with Beta test versions of QDT for a while I found myself looking for problems. This, of course is the job of the beta tester - to break things.

Too Many Betas to the Bar

OK, so I was risking my arm a bit here because Jim was firing new versions of QDT at us on a regular basis during January and I also took on testing QPC v3.30 and QPCPrint for Marcel. Running three beta test versions of new programs is asking for a headache and I got one (and passed it on to Marcel and Jochen).

A bit of background here. My invoicing program for QBranch is written in compiled BASIC using DATAdesign and pfDATA. The lists of products are just plain text lists read into the program when needed. QPC2 v 3.30 was ready to fly so I entered the text file into QD and edited the version number in the list and saved it. When I made the next sale the file was corrupt. Now during the testing phase I came across an instance where menus were corrupting and was never quite sure how to pin that down. I could reproduce the problem here and on two machines but

I could not make it do the same on Marcel's. All I was doing at the time was removing the lines which fire up QDT.

This should make no difference to the bulk of the BOOT file because the whole of QDT is configured and started by 10 or so lines at the end of the BOOT file. My previous BOOT file worked perfectly and, during the installation of QDT all that happens to the BOOT file is that the installer adds the lines to fire up the program. Jim needed me to do a clean install so I deleted the QDT folder, removed the lines from the BOOT and QPC.

When QPC next appeared certain sections of the Menu Extensions seemed to fail. Had this been a PC system I was fiddling with I would have assumed the Registry was corrupt or the system missing a dll but one of the joys of the SMSQ/E environment is the very robustness of the system. If you load an extension it is there and if you don't it isn't. The only problems you may get are when a program calls an extension which is not there. Then, if it is written correctly, it should just report the lack of the extension and exit leaving the system intact. Of course this also has a few drawbacks in other ways but on the whole it is a very flexible system.

The odd thing was that I could add the QDT lines to the BOOT file and it all worked again. I did this several times during that week and it had exactly the same effect. Not only that but all I had to do was to LRESPR the same copy of Menu_Rext and it was OK too. As I said we never quite got to the bottom of that so, when the corrupted file appeared I assumed I had hit another manifestation of the same effect. The problem was that QD was

compressing the TABs and this made the file unreadable in my program. I could reload menu_rext and they were not compressed. I checked the configuration of QD but could see nothing there and I knew I had not changed it for ages. Marcel gave me the answer through a reference to the TAB settings in QD. Of course it was not QD that had changed but Menu_Rext! One of the configurable items in the Menu Extensions is the list of file extensions it uses. I had changed the list when realised it was more convenient to have _TXT as the first item rather than _ASM which is how it is shipped. QD takes its TAB compression settings from the extensions of the file it is loading but, crucially, the list is ordered by the setting in the menu extensions (still with me?) I changed the order of the list in Menu_Rext and QD compressed _TXT files. A rare case where changing one item in one program will affect a completely different one.

CD Confusion it Causes

I have lavished a certain amount of praise on both Thierry Godefroy and Duncan Neithercut in the past and that has been wholly justified but another of the month's upsets crept as a result of the CD reading extensions/program provided by these two for the Qubide and Q40. I am somewhat writing in the dark here because, by the time the ability to read CDs on a Native QL system had come to be I had pretty much moved over to running everything on QPC2. I did make a half hearted attempt at getting them working when they first came out but the Q40 did so little work that it did not seem worth the expense of buying a drive for it.

These days 52 speed CD ROM drives cost less than a floppy drive did back then so I may try it again.

Brian Kemmett, a long time QBranch customer was one of the first to take up QDT, and, also, one of the first to report an error. One problem both he and David Gilham found was that they could not read the QXL.WIN files on the CD. They were using a Q40 and the CD-Eze software for this but they should still have been able to access the files. The CDs read perfectly well on all of the PCs I tried them on and on other versions of QPC2. These were made and written with exactly the same drive as I used on the QL Today CD a while back so I cannot understand why that should be.

They were, however, able to retrieve the information from the rest of the CD and used this to install. Brian then reported that the background files would not work from QDT although he could load them using a BASIC command. Now this was odd because it is the same BASIC command that QDT uses to load them. Of course this was another red fishy thing. By telling us that the file would load normally from outside the program we had our minds locked on a problem within the program. It was only when we asked about the file names themselves that we twigged the answer.

The CD reading program truncates the file name at the 36 character mark and this truncation lops off the extension which Jim's routine was looking for to identify the file type. a swift bit of renaming and it all worked OK. Now I don't know what it is that causes the names to be truncated but there should, at least, be a warning that it has happened

to allow the user the chance to correct the problem or, more sensibly, truncate the name in the middle and leave the extension intact. That would have worked in this case.

Fassssssttttt!!!!!!

A recent posting to the QL Users list mentioned a super fast QL emulator, Qemufast, written by Daniel Tendina, author of QEmulator. You can get this from:

<http://users.infoconex.com/daniele/win/QemuFast.zip>

This would seem to be a lot quicker than anything we have seen before. This got many peoples' ears straining at the alert but was tempered by the assertion that it achieved its main speed boost at the expensive software compatibility. Now to me that is like saying 'I have a train that can go faster than any other because it only stops at two stations'. All very well if these are the two stations that you use but what is the point otherwise. Of all the things that the QDOS/SMSQ user needs speed is hardly near the top of the list. Considering that most QL emulators achieve speeds many times in excess of any of the native hardware I was a bit bemused to see this appear. I am not knocking the expertise that has been used to create this at all but I would be very interested to find out to what use anyone has put it.

Many of the things that are needed are being tackled in one way or another and it does amaze me how well we all work together sometimes. One thing we get little word on is how the other Emulators are developing and that is somewhat disappointing. If there is more development work going on out there than we see in

Kaiser-Wilh.-Str. 302 D-47169 Duisburg
Tel. 0203 502011 Fax 0203 502012
http://smsq.j-m-s.com smsq@j-m-s.com

QDT - The QL DeskTop

QDT brings the QL a modern, easy- and intuitive-to-use graphical user interface. Arrange your programs, games, applications in folders, start your favourite applications with a single click. Fully configurable!

QDT comes with an automatic installer – like a guided tour. Installing it is as easy as never before!

System requirements:

- SMSQ/E Version 3.06 or later
- Graphics Driver 2 Colour support
- HARD DRIVE with at least 3MB free space.
- 4MB RAM

What do you get when you order QDT?

- A floppy disk for Aurora or quick-start users.
- A CD ROM containing a QXL.win file, backgrounds, sprite editor, and many other extras.
- A printed quick-start guide - 12 months free updates!

QDT costs EUR 43,- plus postage (EUR 3,- Europe, EUR 6,- rest of the world).

NEW! NEW! NEW!

Easy Pointer Version 4

NEW!

You have probably read a lot about EasyPTR V4 in this issue - now is the time to order it!

Upgrade from old versions of EasyPTR EUR 39,90 plus p&p

Brandnew version, if you do not own any old version of EasyPTR EUR 59,90 plus p&p

(Postage EUR 5,- Europe, EUR 8,- rest of the world)

BALLED! - new game for the Pointer Environment

Balled is the new Game from Wolfgang Lenerz for QDOS and SMSQ/E. It runs in high-colour mode (looks great) and can also be played in standard QL Mode 4.

Also, Balled uses the digital sound system if you run it on a recent version of SMSQ/E and QPC, Q40 and Q60! The aim of the game is to build lines by moving coloured balls around, following some rules, of course. After every move, more balls appear, so you have to be careful not to fill up the board – then the game is over! Different levels of difficulty and wildcard balls are also featured. Find out how to get bonus and higher scores to fill up the high-score table!

Balled costs only EUR 11,90 plus postage (EUR 2,- Europe, EUR 4,- rest of the world).

QPCPrint - printer driver for QPC

NEW!! NEW!!

NEW!

QPCPrint allows you to print to virtually any printer connected to your PC running QPC, even to fax and pdf printer drivers. It accepts EPSON ESC/P2 codes to any PAR printer and converts it to output which can be handled by Windows, which looks very similar to original EPSON output. You can even configure the individual fonts used by the emulation. This is THE application many QPC users waited for a long time! Shipped on CD.

QPCPrint costs only EUR 39,90 plus postage (EUR 3,- Europe, EUR 6,- rest of the world)

If you wish to place your order via internet, please do not send your credit card details in an email!

Please use the secure contact form on the SMSQ homepage: SMSQ.J-M-S.COM

We accept VISA, MasterCard and Diners Club!

the pages of this magazine, then I would like to extend an invitation for those with experiences of the other emulators to come forward and speak. I realise that I may seem as though I am all for the monotheistic world of SMSQ/E but I do not want these other users to be out in the cold. Indeed I think it is vital that they are in the forum and communicating with each other because my main issue with the control of SMSQ/E was compatibility. I would like that compatibility to extend across emulations too. We are all using the same software after all.

Amazing What You Can Find When You Are Not Looking

I was looking back through the QL Users Archive trying to find the name of the fast QL Emulator but could not quite track it down so I did a Google search which led me to a gamers mailing list. There was a discussion of Spectrum emulators and such and this comment:

'Alright fair enough you win. Persuade someone to create or port over a QL emulator for the sake of 3 or so games and i'll be impressed. There's many other machines (Eg Oric / Dragon) which could be ported which have a far better range of games than the QL. I'm a huge Speccy fan but i don't think a QL emulator would be that good or hardly used by many people'

Now did you ever hear the sound of a glove hitting the floor? I mean, really! I know many of you do not use your systems for game playing but there were quite a few good games for the QL (have a look at RWAP Software' web pages and catalogue if you don't believe me) and it was better

than the Spectrum by a long streak of white writing stuff. I just had to subscribe to that list to reply.

I have had no response to my posting as yet but I will let you know if anyone comes back whistling in appreciation.

QPC2, QPC2 On a Stick.

Not an Albatross this one but Ken Bain's little mention of how he carries a complete QL system around on his 512Mb USB Stick got a few people very excited. Many of us have a similar thing and I have carried my main QXL.WIN-file around on one of these for ages but his novel way of setting up QPC2 so it will work (see elsewhere in this magazine) on any machine was very inventive. Of course this opened the floodgates and we had people with QPC systems on Compact Flash, Secure Digital and even, ironic this one, an IBM Microdrive (The disk comes full circle). One inventive soul had his system on a Sony MP3 player. You can say what you want about QL Users we are very versatile. There are not many systems that you can carry around on a 256mb USB stick are there?

Some time ago I mooted the idea of QPC2 on a PDA and was told that PDA screens had too low a resolution and the minimum resolution for a QL emulation was 512 x 256. I do notice now that the latest iPqcs have larger screen resolutions that. So who is going to be the first with a PDA QL then?

You Easily Pointing At Me?

I decided, since Marcel had sent me the patched EasyMenu program which will work

with the new colours. That the time had come to update the Q Branch Office program. It was a matter of moments to redesign the menu to use the new colours but when trying to substitute high colour icons for the old four colour ones I ran into problems. Marcel never said that this was a 'fixed' program, only that he had applied a few changes to it. I emailed him to ask if he knew of the problem and of solution and the answers were 'yes' and 'no' respectively.

That great serendipity machine in the sky kicked in and without my intervention (honest) other users began asking if there were updated versions of EasyPtr.

Now we had discussed this at the big Eindhoven show and Marcel was adamant that he did not want to get involved in revising EasyPtr. He felt the code was too sparsely commented and idiosyncratic in its construction. I can accept this but many people see Marcel as the 'last great programming hope' ('I could have been a cancantator') and wanted more.

A brief mention in the QL Users list (not by me I hasten to add) brought a quick response for those who still use the program. This was more to do with people asking about programming menus in high colour than with updating Easyptr but it did bring the discussion into the open. The only other way to program menus for the Pointer Environment is to write physical lists of what goes where and link those into the program as in QPTR. Now I never found that a very acceptable solution and EASYptr, for all its drawbacks, and there are a few, is simple and elegant. Using Easymenu you can see exactly what you are doing.

Per Witte and a couple of others were also very vociferous in their support for this. Some of the users had come up with solutions to get around the various shortfalls but it was obvious that a hard core of users wanted an upgrade.

Dragged by the Shirt Tails

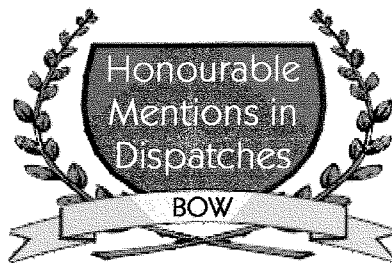
Marcel started off at a brisk pace in the opposite direction from this discussion but they had him by the shirt tails and before he realised it he had written 'well, okay I will look at it but...' followed by both feet into the bear trap of 'in the next release it will...'. It was all over at this point 'Release!!!?' The pack were salivating now and their quarry was trapped.

So now, after a lot of hard work and several statements like 'I know I said I wouldn't do this but...' here I am beta testing EasyPTR v4. I cannot heap too much praise on this because I am one of the traders who is selling it but I do have to say that Marcel has done an excellent job and taken the task far beyond my expectations. I mentioned to him, in a private email after the fifth update had landed in my inbox for testing, that for something he said he was not going to do he was very productive. His reply was

'That's mostly why I didn't want to do it, being a perfectionist I can sometimes be carried away ;-) It's much more now than I ever intended. But I do hope it's good enough now.'

Good enough? It's marvellous! The best way to thank him for all this work is with a flurry of programs that use it. I am not talking about commercial stuff here - although we could do with some nice applications

but some free or shareware stuff would be good. You now have the means to make simple programs look great. What are you all sitting there reading this for? get coding! (but not till you have finished reading my column and filled in your re-subscription forms)



Too many revolutions in a short time can leave you head spinning slightly but Marcel managed to cap off a memorable period in QL history with his remarkable QPCPrint program. This really is the complete solution to printing from QPC2 even though it cannot be utilised on other systems. It is smooth, elegant and, crucially, requires very little user intervention to get great results.

Here at Q Branch Towers there are three printers. An ancient Epson Stylus 600 which I bought from Bill Waugh a few years back is the only colour printer here which is 'QL Compliant' and got dragged around to shows as well as serving duty as the main Q Branch QL printer. The Epson C80, which I bought to replace it, has only ever worked correctly from Windows and remained attached to my main PC. I added the EPSON EPL-6200 laser printer for doing manuals and the like and this will work fine from both QPC2 and Windows. When I got the first Beta test version of QPCPrint I found I was able to print to the C80 from QPC2 with no hassle. This was a real red letter day here. No longer did I have to set up the laptop so I could

print one invoice. I could now do it all from the desktop machine. The thrill of being able to do this was augmented by the way that QPCPrint works. On the laptop I had to produce the invoice and print it and then wait till the printer had accepted all of the output before I could do anything else. While it was feeding the printer everything was a little jerky and hard to control. With QPCPrint the program produces the job and sends it off to the Windows spooler to print it. Once that is happening you have control of the computer again and can get on with the next one. There is a pause while the program does the conversion in Windows but that does not affect the SMSQ/E side of things and the printout is a little delayed but the added bonus is that it comes out of the machine really quickly.

Of course this sudden rise in use of my C80 caused the printer to throw a spasm so I began to look for a replacement unit. It then I realised I could have any printer I could afford! No more trailing through the manufacturers specs for QL compatibility. Oh frabjulous day!! I wound up choosing a Canon MP750 which is a great printer and also a scanner and copier with a features list as long as your arm. Not only this but the consumables are cheap and the print heads are user replaceable. Once I had this installed in windows QPC just printed to it. Now this is real progress.

So, Marcel, another feather in your cap for this one and you must have the most Honourable Mentions of everyone. Well done. Now about that walking on water you have been doing.....



The QL Show Agenda



QL2004 AFTERGLOW

International QL Meeting - (NL) Eindhoven

Saturday, 18th of June, 10:00 to 16:00

Pleincollege St. Joris, Roostenlaan 296

We hope to turn this QL show into another major event!

All major QL dealers will be there!

Reserve this date in your agenda and come to the show - we would really like to repeat the success of QL 2004!

The Quanta "QL is 21" Show

International Meeting - (UK) Portsmouth

Sat., 29th of Oct. and Sun., 30 th of Oct.

Ibis Portsmouth Centre, Winston Churchill

Avenue, PO1 2LX Portsmouth.

Sunday will be a short day to allow most people to drive home before it is dark.

Directions: From M27 turn off at Junction 12 onto M275 and then follow signs for city centre, sea front and then Guild Hall. For public transport users Portsmouth and Southsea station is 5 - 10 minutes walk from the hotel.

For people needing overnight accommodation in the hotel there is a special bookings procedure. Details from Quanta.

There will be both an activity programme and a programme for partners. Shortly after the Quanta AGM we should have a website available giving more details.

Last-Minute News: NASQLUG is pleased to announce 2005 US QL show in Fitchburg, Mass.; 25 miles west of Boston. Site near Fitchburg Econolodge. Exact site and times to be announced soon. In the meantime, attendees can make reservations and room selection at www.econolodge.com at \$49.90/night.

Fitchburg is the prettiest place on earth in the fall due to the change of fall leaf colors. It is something everyone should see at least once in their lifetime. Fitchburg is at the east end of route 2 and is one of the prettiest scenic drives in America. It runs through the Massachusetts Berkshires with breathtaking views.